

# **Pedestrian Flow Models**

**Raghavender Etikyala**

Vom Fachbereich Mathematik  
der Technischen Universität Kaiserslautern  
zur Verleihung des akademischen Grades  
Doktors der Naturwissenschaften  
(Doktor rerum naturalium, Dr. rer. nat.)  
genehmigte Dissertation.

1. Gutachter: Prof. Dr. Axel Klar
2. Gutachter: Prof. Dr. Simone Göttlich

Datum der Disputation: 13 May 2014

D386



## Acknowledgements

The life of a PhD student has never been easy but I have certainly relished this experience. During this period I derived my inspiration from several sources and now I want to express my deepest gratitude to all those sources.

Foremost, I owe thanks to my supervisor Prof. Dr. Axel Klar who has not only encouraged me but has given his remarkable suggestions and invaluable supervision throughout my thesis. His support is invaluable.

Now I would like to express my sincere gratitude to Dr. Sudarshan Tiwari for his advices, continuous encouragement and support right from the start to the end which have always been the driving force towards the successful completion of my thesis.

I express my sincere appreciation to Prof. Dr. Simone Göttlich for serving as a reviewer of my doctoral thesis.

I would like to thank Dr. Falk Triebisch for his continuous help throughout the scholarship period.

I am very grateful for the fruitful working environments provided by the Department of Mathematics at the University of Kaiserslautern. I am also very grateful to RHRK at the University of Kaiserslautern for providing me the high performance cluster Elwetritsch for my computations. I like to thank to my colleagues and friends in the Technomathematics group with whom I had a wonderful time and had very helpful discussions.

Now I thank the financial support I received from DAAD (Deutscher Akademischer Austausch Dienst) for the pursuit of this research work.

Finally, I would like to express my deep obligation to my parents and relatives back home in India. Their consistent mental supports have always been the strong push for me during this period.





# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Motivation . . . . .	1
1.2	Overview . . . . .	3
1.3	Outline of Contents . . . . .	6
<b>2</b>	<b>Hierarchy of Pedestrian Flow Models</b>	<b>9</b>
2.1	Overview . . . . .	9
2.1.1	Microscopic . . . . .	9
2.1.2	Macroscopic . . . . .	16
2.2	The Microscopic Model . . . . .	18
2.3	Mean Field Equation . . . . .	22
2.4	Hydrodynamic Models . . . . .	24
2.5	Scalar Models . . . . .	28
<b>3</b>	<b>The Eikonal Equation</b>	<b>31</b>
3.1	Introduction . . . . .	31
3.2	Existing Methods . . . . .	33
3.2.1	Upwind Approximation of the Eikonal Equation . . . .	33
3.2.2	Fast Marching Method . . . . .	34
3.2.3	Fast Sweeping Method . . . . .	38
3.3	Finite Pointset Method . . . . .	39
3.3.1	If cardinality of the set $N_{xa}$ is 1 . . . . .	40
3.3.2	If cardinality of the set $N_{xa}$ is 2 . . . . .	40

3.3.3	If cardinality of the set $N_{xa}$ is greater than 2 . . . . .	42
3.3.4	Algorithm . . . . .	45
3.3.5	Time Complexity . . . . .	46
3.3.6	Numerical Examples . . . . .	46
<b>4</b>	<b>Numerical Method</b>	<b>55</b>
4.1	Microscopic Simulation . . . . .	55
4.2	Hydrodynamic and Scalar Simulation . . . . .	56
4.3	Finite Pointset Method (FPM) . . . . .	57
4.3.1	Least Square Approximation . . . . .	58
<b>5</b>	<b>Numerical Results</b>	<b>63</b>
5.1	Numerical Example 1 . . . . .	63
5.1.1	Initialization . . . . .	63
5.1.2	Boundary Conditions for the Eikonal equation . . . . .	65
5.1.3	Parameters . . . . .	65
5.1.4	Results . . . . .	66
5.2	Numerical Example 2 . . . . .	78
5.2.1	Initial and Boundary Conditions . . . . .	81
5.2.2	Parameters . . . . .	82
5.2.3	Results . . . . .	82
<b>6</b>	<b>Conclusions</b>	<b>95</b>

# Chapter 1

## Introduction

A general overview of the present work is discussed in this chapter. We start with the motivation behind the pedestrian modelling. We then proceed to the general overview of pedestrian models along with the Eikonal equation. A brief outline of the thesis is also given in this chapter.

### 1.1 Motivation

In recent years, research on pedestrian flow has become more popular and has attracted the interest of an increasing number of scientists. The modelling of pedestrian behaviour in a real-world environment is a complex problem, mainly due to the unpredictable nature of human decision making. Knowledge of pedestrian decision-making and movement is critical in a variety of application domains.

Pedestrian models have many applications. For example, understanding of pedestrian movement such as lane formation in uni-directional or bi-directional flow, interaction with other pedestrians, attraction towards any entertainment, behaviour near corners and in panic situations is important for planning and designing public buildings such as train stations, shopping malls, airports and theatres in terms of the capacity of such buildings and also with respect to issues as safety, evacuation and navigation. While planning, the architectures of these buildings might be interested in how people move around their intended design so that they can place shop entrances, corridors, emergency exits and seating in useful locations. The dimensions



(a) Crowd in Haneda airport, Japan.



(b) Pedestrians in a Mall in Hamburg, Germany.

Figure 1.1: Pedestrian flow in public buildings; Source : (a) [knoxnews.com](http://knoxnews.com) (b) [scabee.com](http://scabee.com)

of public space influence the pedestrian movement and in turn have an important impact on the general atmosphere in pedestrianized areas. Foot traffic is another area where pedestrian models have an application. Analysing the pedestrian behaviour in city center and in busy areas or during office hours is not only helpful in controlling foot traffic but also helps to place the stores depending on the pedestrian movement patterns.

In addition to the above applications, pedestrian movement patterns are critical in planning the large mass gatherings such as religious meetings and new year celebrations. There have been many crowd disasters in recent times because of poor planning. Few of the crowd disasters are presented in Table 1.1, more can be found in [77].

The exact knowledge of pedestrian behaviour plays an essential role in the planning of big events, where high densities of pedestrians are expected. The organisers require information on what areas are likely to be congested so that management strategies can be developed and tested before event. It is difficult to control and predict the dynamics of a large number of pedestrians, especially if panic appears and dominates the reactions of a crowd. The main reason for any tragedy is mistakes in the planning phase for the event and a combination of events leading to the development of local panic. Especially in evacuation scenarios one needs to know the routes of the pedestrians to avoid jams and potential injuries. Many accidents where crowd of pedestrians were involved, happened because of planning mistakes and not because of mass panic.



Figure 1.2: Pedestrians on streets; Source: (a) [chicagotribune.com](http://chicagotribune.com)  
 (b) [farmeronamission.blogspot.com](http://farmeronamission.blogspot.com)

Experiments are one way to know the behaviour of pedestrians in various situations but experiments with many people are expensive and simulating experimentally stress situations like crowd disasters is not possible. This is where pedestrian models come into picture. Pedestrian models are useful in analysing various situations through simulations.

## 1.2 Overview

In recent years a large number of models for pedestrian flow have appeared on different levels of description from microscopic to macroscopic. On the microscopic (individual based) level, the available models are benefit cost cellular model [32], cellular automata models [6, 7, 8], magnetic force model [63], queuing network model [86] and Social force model [35, 38]. Brief introduction of these models is given in chapter 2.

Equations on the mesoscopic or kinetic level are discussed in Ref. [36, 25]. Hydrodynamic pedestrian flow equations involving equations for density and mean velocity of the flow are derived in Refs. [36, 5]. In Refs. [43, 44, 31, 2] modeling of pedestrian flow with scalar conservation laws coupled to the solution of the Eikonal equation has been presented and investigated. The first modeling attempt is due to Hughes [43] who defined the crowd as a thinking fluid and described the time evolution of its density using a scalar conservation law. Gas dynamic approach to the pedestrian flow is presented by Bellomo and Dogbe [5] and Maury et.al [59] presented gradient flow type

Year	Place	reason	# deaths	# injured
2000	Denmark	Crowd surge and crushing in mosh pit area	9	-
2001	South Africa	Overcrowding	43	-
2001	Japan	Stampede after firework show	11	247
2004	China	Stampede to see fireworks display	37	15
2006	Saudi Arabia	Stampede while throwing stones at pillars	345	-
2010	India	Stampede during religious event	71	200
2010	Germany	Overcrowding	21	510
2010	Cambodia	Swaying bridge causing panic	347	-
2013	India	Roumer of bridge falling	115	110
2014	India	Religious stampede	18	40

Table 1.1: Crowd disasters; Source : [en.wikipedia.org/wiki/List-of-human-stampedes](http://en.wikipedia.org/wiki/List-of-human-stampedes)

model for pedestrian flow in macroscopic level. Finally, scalar equations including a nonlocal interaction between pedestrians are presented in Refs. [16, 17].

For a general recent review on pedestrian flow models we refer again to Refs. [4, 21]. For further references, in particular, for the derivation of kinetic and continuum models from microscopic models see Ref. [5] and also the discussion in Ref. [21].

We choose a classical microscopic social force model for pedestrians Ref. [40] and extend it with an optimal path computation as, for example, in Ref. [43]. Thus, additionally to the local interaction between pedestrians, a non-local term including a global knowledge of the physical setting is introduced. These equations are approximated using a scaling assumption, see Ref. [11], and a mean field equation with a convolution term is derived from the local interaction in the microscopic model. This procedure is classical, but is in contrast to the kinetic equations obtained, for example, in Refs. [36, 25], where classical Boltzmann terms are used for the modelling on the kinetic level. Derived models based on microscopic models with constant velocity



(a) Crowd at macca



(b) Crowd at Love parade dance festival in Duisburg

Figure 1.3: Crowd gatherings; Source : (a) [ramadan-karim.com](http://ramadan-karim.com) (b) [rnw.nl](http://rnw.nl)

are discussed as well. We refer to Ref. [21] for a model hierarchy based on a microscopic constant velocity models and a mean field approach for the so called heuristic behavioural individual based model.

The usual hydrodynamic limit assumptions lead to macroscopic equations for the density and mean velocity of the pedestrian flow, compare Ref. [11] for the case of swarming models. The hydrodynamic equations obtained here still contain a nonlocal interaction term due to the derivation from the mean field equations with non local interaction terms. Finally, further simplifications yield scalar limit equations with a non local term and an Eikonal equation as in Refs. [16, 17, 18] and [43, 44]. Moreover, a local Eikonal equation to model a local visibility of the flow for each pedestrian is discussed. This can be viewed as an intermediate model between the global view via the global solution of the Eikonal equation and the local interaction terms. Such an approach requires a solution of Eikonal equation in a neighborhood of the actual particle and is not pursued here further.

For the numerical simulations we use particle methods on the microscopic and macroscopic level of the model hierarchy. These methods are straightforward for microscopic equations. In case of the macroscopic equations particle methods are based on a Lagrangian formulation of these equations. One approximates the spatial derivatives at the particle locations by a least squares difference approximation. The integral over the interaction potential is evaluated by a straightforward integration rule or in case the numerical simulation is underresolved by a higher order approximation of the integral. The models and methods are compared from the point of view of qualitative

behaviour and computation time and the advantages of using macroscopic models and the associated particle methods compared to a microscopic simulation is pointed out. We refer to Ref. [83] for a related but simpler approach.

We present and use a mesh free method to solve the Eikonal equation. The most popular methods to solve the Eikonal equation are fast marching methods [73, 74] and fast sweeping methods [92, 47, 87]. In realistic grids and complex geometries fast marching method is best suited than fast sweeping method in terms of time complexity, see Ref. [33]. However, fast marching method is only applicable for orthogonal grids and triangular grids. In case of triangular grid, it is complicated to use when the triangles are not acute. We present a mesh free method, which follows the idea of front marching in fast marching method but differs from it in approximating derivatives in the Eikonal equation. We use the least square method to approximate the derivatives. Our method is simple to implement and can be applied on any arbitrary grid and in complex geometries.

The parts of the present work have been published in [28, 29, 52]. Particle methods for pedestrian flow models from microscopic to non-local continuum models is published in [28]. Mesh free method to solve the Eikonal equation is published in [52] and a macroscopic model for pedestrian flow and comparison with the experimental results is published in [29].

### 1.3 Outline of Contents

Chapter 2 presents the hierarchy of pedestrian models from microscopic to non local continuum. In Section 2.1, we give brief overview of the existing models in microscopic and macroscopic cases. We present a microscopic model with optimal path computation using the Eikonal equation in Section 2.2. In Section 2.3, we derive the mean field equation and hydrodynamic model is presented in Section 2.4. Section 2.5 provides the derivation of scalar model.

Chapter 3 discusses about the Eikonal equation. We start with a brief introduction of the Eikonal equation in Section 3.1. In Section 3.2, existing methods to solve the Eikonal equation are discussed briefly. In Section 3.3, a mesh free method to solve the Eikonal equation along with the algorithm is given. Time complexity and few numerical examples to validate our method are also presented in Section 3.3.

In Chapter 4, we explain the numerical methods to solve the governing equa-



tions in the microscopic, hydrodynamic and scalar models. The finite point set method is presented in Section 4.1 .

In Chapter 5, we present two numerical examples. In first numerical example we investigate the microscopic, hydrodynamic and scalar models. We consider the example of pedestrian flow in a railway platform with obstruction as a test case and compare microscopic, hydrodynamic and scalar models. We present various patterns for different obstructions and for different parameters for this example. In the second numerical example we are interested in comparing our results with the experimental results. We consider the experimental results of uni and bi directional pedestrian flow in straight corridor and pedestrian flow through T-junction and we compare our results with the experimental results.

Finally, Chapter 6 concludes the present work.



# Chapter 2

## Hierarchy of Pedestrian Flow Models

In this chapter, we start with an overview of the existing models for pedestrian flow in micro and macro scale. Then a hierarchy of models for pedestrian flow is presented. It includes microscopic models based on interacting particle system coupled to the Eikonal equation, hydrodynamic models using equations for density and mean velocity, nonlocal continuum equations for the density and diffusive Hughes equations.

### 2.1 Overview

In this section, we give an overview of the existing pedestrian models on microscopic and macroscopic scales.

#### 2.1.1 Microscopic

In microscopic models pedestrians are represented by their position and velocity. Most of the mathematical models at the microscopic scale have a structure similar to that of Newtonian dynamics. The structure of models is as follows:

$$\begin{aligned}\frac{dx_i}{dt} &= v_i \\ \frac{dv_i}{dt} &= F_i(x_1, \dots, x_N, v_1, \dots, v_N),\end{aligned}$$

where  $x_i$  is the position,  $v_i$  the velocity and  $F_i$  the total force of  $i^{th}$  particle. The solution of the above system provides the time evolution of position and velocity of pedestrians. Different modeling approaches correspond to different ways of describing the acceleration term on the basis of a detailed interpretation of individual behaviors.

### Benefit Cost Cellular Model

This model is proposed by Gipps and Marksjo [32]. It simulates the pedestrian as particle in a cell. The computational domain is divided into rectangular cells with each side equal to  $0.5m$  and each cell can be occupied by at most one pedestrian and a score is assigned to each cell on the basis of proximity to pedestrians. This score represents the repulsive effects of nearby pedestrians and has to be balanced against the gains made by the pedestrians in moving towards his destination. When the fields of two pedestrians overlap, the score is the sum of the score generated by each pedestrian individually.

In this model, the pedestrians do not take account of the directions in which other pedestrians are travelling, the scores would be symmetric around a pedestrian. The cell occupied by a pedestrian is given a score of 1000, the cells with a side in common a score of 40, the cells with a corner in common a score of 13 and lower values to the more distant cells. The very high repulsive score in the cell occupied by a pedestrian serves as barrier to another pedestrian joining him in the cell. The scores in the surrounding cells are approximately inversely proportional to the square of the separation of pedestrians in the two cells. The formula for repulsive force is  $\frac{1}{(\Delta-\alpha)^2+\beta}$ , where  $\Delta$  is the distance separating the centres of the two cells,  $\alpha$  is the constant less than the diameter of the pedestrian and  $\beta$  is an arbitrary constant to moderate fluctuations in scores.

If the measure of the gain obtained by moving to a particular cell is a function of the change in separation of the pedestrian from his destination, the longer step length associated with diagonal moves will cause the pedestrian to move diagonally in the absence of other pedestrians. The tendency towards diagonal moves can be overcome by making the gain from move dependent on the angle of deviation from the desired path. The function for the gain has to be positive when the deviation is less than 90 degrees and negative when the deviation is greater than 90 degrees. The formula for the gain is  $K.\cos(\sigma).|\cos(\sigma)|$ , where  $K$  is a constant of proportionality to enable gain

of moving in a straight line, to be balanced against the cost of approaching other pedestrian closely and  $\sigma$  is the angle by which the pedestrian deviates from a straight line to his destination. From elementary geometry, we have

$$\cos(\sigma) = \frac{(x_i - s)(d - s)}{|x_i - s||d - s|} \quad (2.1.1)$$

where  $x_i$  is the location of target cell,  $s$  is the location of the pedestrian and  $d$  is the location of the destination. The formula for net score is given by subtracting the cost of moving closer to other pedestrians from the gain the pedestrian obtains by moving closer to the destination.

$$\text{Score} = \frac{K(x_i - s)(d - s)|x_i - s||d - s|}{|x_i - s|^2|d - s|^2} - \frac{1}{(\Delta - \alpha)^2 + \beta}. \quad (2.1.2)$$

The cell which the pedestrian moves is determined by calculating a net benefit for the move to each of the accessible cells and selecting the cell with the maximum benefit. The pedestrian has to select the cell with the maximum net benefit from the set of nine neighboring cells.

Being simple is the advantage with this model but the model has a drawback due to the arbitrary scoring of the cells and the pedestrians. The scoring system makes the model difficult to be calibrated with the real world phenomena.

### Magnetic Force Model

This Model is developed by Okazaki and Matsushita [63]. Pedestrian movement is caused by the application of magnetic models and equation of motion in the magnetic field. Each pedestrian and obstacles such as walls have positive pole. The destination of the pedestrians has negative pole. Pedestrians moves to their goals and avoid collisions. Two forces work on each pedestrian. First, magnetic force as formulated by Coulomb's law, which depends on the intensity of magnetic load of a pedestrian and distance between pedestrians, given by

$$F = \frac{kq_1q_2r}{|r|^3}, \quad (2.1.3)$$

where

- $F$  is a magnetic force vector
- $k$  is a constant

- $q_1$  is intensity of magnetic load of a pedestrian
- $q_2$  is intensity of a magnetic pole
- $r$  is a vector from a pedestrian to a magnetic pole

Another force, which acts on a pedestrian to avoid the collision with another pedestrian or obstacle. For example, in Figure 2.1 where Pedestrian  $i$  tries to avoid the collision with pedestrian  $j$ , the force exerts acceleration on pedestrian  $i$ . Acceleration is calculated by

$$a = v_i \cdot \cos(p) \cdot \tan(q), \quad (2.1.4)$$

where

- $a$  is the acceleration acting on pedestrian  $i$  to modify the direction of  $v_{ij}$  to the direction of line  $ik$ .
- $v_i$  is the velocity of pedestrian  $i$
- $p$  is the angle between  $v_{ij}$  and  $v_i$
- $q$  is the angle between  $v_{ij}$  and  $ik$
- $v_{ij}$  is the relative velocity of pedestrian  $i$  to pedestrian  $j$ .

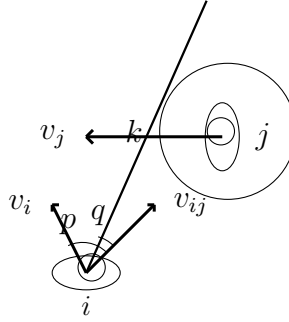


Figure 2.1: Additional repulsive force on magnetic force model.

In Figure 2.1, acceleration  $a$  acts on pedestrian  $i$  to modify the direction of  $v_{ij}$  to the direction of line  $ik$ . Line  $ik$  is a contacting line from the position of pedestrian  $i$  to the circle around pedestrian  $j$ . Totality of forces from goals, walls and other pedestrians act on each pedestrian and it decides the velocity of each pedestrian each time.

As in the benefit cost cellular model, this model also has the same problem of arbitrary setting up of the magnetic intensity values. Due to those arbitrary setting of the magnetic load, the validation of the model can only be done merely by visual inspection. No real world phenomena can be validated using this model.

### Cellular Automata Model

Cellular Automata models have been applied for simulating car traffic and validate adequately with the real traffic data. Recently, cellular automation model has been used for pedestrians [6, 7, 8].

The model simulates pedestrians as entities in cells. The walkway is modelled as grid cells and a pedestrian is represented by a circle that occupies a cell. The grid has to be regular, which means that it is made of any sort of regular polygons. Three possible types of grids are triangular, rectangular and hexagonal grid. Since pedestrians are represented by circle, hexagonal grid is best choice of grid cell. But it is difficult to implement in computation and also it is not ideal to represent straight walls and obstacles. Similarly, triangular grids are also not suitable for straight walls but best suited in representing complex geometries. A rectangular grid is perfectly suited for walls, because most rooms are rectangular.

The computational domain is divided into cells. Each cell can be empty or occupied by exact one pedestrian. The size of a cell corresponds to approximately  $0.4m \times 0.4m$ . The update is done in parallel for all particles. Each pedestrian is given a direction of preference i.e., his destination. From this direction a  $3 \times 3$  matrix of preference is constructed which contains the probabilities for a move of the pedestrian. The central element of the matrix describes the probability for the pedestrian to take no move at all, the remaining eight correspond to a move to the neighboring cells. The probabilities can be related to the velocity and the longitudinal and transversal standard deviations. In each time step, for each pedestrian a desired move is chosen according to these probabilities. This is done in parallel for all pedestrians. If the target cell is occupied, the pedestrian does not move. If it is not occupied and no other pedestrian targets the same cell, the move is executed. If more than one pedestrian share the same target cell, one is chosen according to the relative probabilities with which each pedestrian chose their target. This pedestrian moves while its rivals for the same target keep their position. The speed to pedestrians is assigned based on the available gap and advanced forward by this speed. A gap is the number of empty

cells ahead. The range of allowable movement is equal to minimum of one of gap or maximum walking speed. Though the cellular automata model is also simple to develop and fast to update the data, the heuristic approach of the updating rules is undesirable since it does not reflect the real behavior of the pedestrian. The inherent grid cells of the cellular based model make the behavior of pedestrians seems rough visually. The pedestrian gives the impression of jumping from one cell to another.

### Queuing Network Model

Queuing Network Model is used for the pedestrian simulation for evacuation purposes [79, 80, 58]. The approach is a discrete event Monte Carlo simulation, where each room is denoted as a node and the door between rooms as links. Each person departs from one node, queues in a link, and arrives at another node. A number of pedestrians move from one node to another in search for the exit door. Each pedestrian has a location goal. Each person has to move from its present position to an exit as quickly and safely as possible. Route, which each person use and the evacuation time is recorded in each node. When a pedestrian arrives in a node, he makes a weighted-random choice to choose a link among all possible links. The weight is a function of actual population density in the room. If the link cannot be used, a pedestrian will wait or find another route to follow. In the source node, a person needs a certain time to react before movement begins, while in the final destination node he will stop the movement process. Pedestrian crossing has a similar goal to the evacuation where the pedestrians have to move from their original position to the other side of the road as quickly and safely as possible. The evacuation time, as one of the performance measurements will be used in the proposed model. The queuing network model has implicit visual interaction. The behavior of the pedestrians is not clearly shown and the collisions among pedestrians are not clearly guaranteed. The first in first out priority rule that is inherent in the model is not very realistic especially in a crowded situation.

### Social Force Model

Social Force Model is developed by Helbing [35, 36, 38]. According to Helbing, a pedestrian is subject to social forces that motivate the pedestrian. The mathematical formulation of the model read as

$$\frac{dv_i}{dt} = \frac{1}{\tau}(v_i^o e_i - v_i) + \sum_j F_{ij}(x_i, x_j) + \sum_k F_{ik}(x_i) + \text{fluctuations} \quad (2.1.5)$$



The first term in the right hand side represents the force that motivates the pedestrian to reach the goal. Each pedestrian wants to reach a certain destination  $x_i^o$  as comfortable as possible. Therefore, pedestrian normally takes the shortest possible way.  $e_i$  is the unit vector in the direction of the destination which is given by

$$e_i = \frac{x_i^o - x_i}{\|x_i^o - x_i\|}. \quad (2.1.6)$$

$v_i^o$  is the intended velocity with which pedestrians tend to move in the absence of interaction. A deviation of the actual velocity  $v_i(t)$  from the intended velocity  $v_i(t) = v_i^o e_i$  due to necessary deceleration processes or avoidance processes leads to a tendency to approach  $v_i(t)$  again within a certain relaxation time  $\tau$ .

The second term in the right hand side of eqn (2.1.5) represents the interaction force of a pedestrian with other pedestrians. The motion of a pedestrian influenced by other pedestrians. In particular, pedestrian keeps a certain distance from other pedestrians that depends on the pedestrian density and the desired speed  $v_i^o$ . Each pedestrian has a comfortable radius. A pedestrian feels increasingly uncomfortable the closer he/she gets to a strange person, who may react in an aggressive way. This results in repulsive effects of the other pedestrians that can be represented by

$$F_{ij}(x_i, x_j) = -\nabla V(B(r_{ij})). \quad (2.1.7)$$

The repulsive potential  $V(B(r_{ij}))$  is a monotonic decreasing function and  $r_{ij}$  is the distance between pedestrians  $i$  and  $j$ .

A pedestrian also keeps a certain distance from borders of buildings, walls, obstacles. He/she feels more uncomfortable the closer to a border he/she walks since he/she has to pay more attention to avoid the danger of getting hurt. The third term in the right hand side of (2.1.5) reflects this repulsive force, which can be described by

$$F_{ik}(x_i) = -\nabla U(\|r_{ik}\|), \quad (2.1.8)$$

where  $U$  is a monotonic decreasing potential and  $r_{ik}$  denotes the shortest distance between pedestrian and wall or obstacle.

The social force model is the best among all microscopic models that has been developed so far. The variables are not arbitrary because they have physical meaning that can be measured.

## 2.1.2 Macroscopic

Macroscopic description treats the crowd as a continuum medium characterized by averaged quantities such as density and mean velocity.

### Hughes Model

Hughes [43] considered a two dimensional connected domain  $\Omega \in R^2$  corresponding to some walking facility. It is equipped with an exit which models the destination of the crowd motion and can contain obstacles. The boundary of the continuum  $\Gamma$ , comprises three segments, inflow boundary  $\Gamma_o$ , exits  $\Gamma_d$  and walls/obstacles  $\Gamma_h$ .

Let  $\rho$  be the pedestrian density,  $u$  be the velocity of the pedestrian flow, thus similar to many physical systems, conservation of pedestrians implies

$$\rho_t + \text{div}(\rho u) = 0 \quad (2.1.9)$$

Hughes defined a potential function  $\Phi(x, y, t)$  and proposed that the motion of any pedestrian is in the direction to this potential, that is, in the direction for which

$$\begin{cases} \hat{\Phi}_x = \frac{-\Phi_x}{\sqrt{\Phi_x^2 + \Phi_y^2}} \\ \hat{\Phi}_y = \frac{-\Phi_y}{\sqrt{\Phi_x^2 + \Phi_y^2}} \end{cases}$$

Therefore,

$$u_1 = f(\rho) \cdot \hat{\Phi}_x \text{ and } u_2 = f(\rho) \cdot \hat{\Phi}_y \quad (2.1.10)$$

where  $f(\rho)$  is the speed. Pedestrians seek to minimize their estimated travel time, but temper this behavior to avoid extremely high densities. This tempering is assumed to be separable such that pedestrian minimize the product of their travel time and a function of the density. Two pedestrians on a given potential must not be at the same new potential as each other at some later time. Thus the distance between potentials must be proportional to pedestrian speed irrespective of the initial position of a pedestrian. Thus,

$$\frac{1}{\sqrt{\Phi_x^2 + \Phi_y^2}} = g(\rho) \sqrt{u_1^2 + u_2^2} \quad (2.1.11)$$

where  $g(\rho)$  is a factor to allow for discomfort at very high densities. The factor  $g(\rho)$  is equal to unity for most densities but rises for high densities. Finally, the governing equations for pedestrian flow,

$$\rho_t - \text{div}(\rho f^2(\rho) g(\rho) \nabla \Phi) = 0 \quad (2.1.12)$$

and

$$g(\rho)f(\rho) = \frac{1}{\sqrt{\Phi_x^2 + \Phi_y^2}} \quad (2.1.13)$$

Hughes' model is useful for describing the movement of large crowds, as a crowd generally moves as an entity and individual differences are less important. In this method, it is assumed that tall pedestrians have a clear view of the operating conditions of the whole walking facility and that short pedestrians obtain such information from/or follow the route decisions of neighboring tall pedestrians. The model intends to describe the aggregate behavior of a homogeneous crowd with a common goal, such as crowd movement in a railway platform or at a sporting event, holy site, or political demonstration, rather than the heterogeneous behavior of individuals.

### Gas Dynamics Approach

This approach is presented by Bellomo and Dogbe [5]. Consider the crowd in a bounded domain  $\Omega \in R^2$  where  $\partial\Omega$  is its boundary. The overall description of the system is delivered by the equation of conservation of mass and equilibrium of linear momentum defined by the following system of partial differential equations:

$$\begin{cases} \partial_t \rho + \nabla_x \cdot (\rho u) = 0 \\ \partial_t u + (u \cdot \nabla_x) u = F(\rho, u) \end{cases} \quad (2.1.14)$$

where  $F$  models the average acceleration that acts over the elementary block of individuals in volume  $dx dy$ . Bellomo et al presented three different classes for various  $F$ .

Class one refers to systems where the pedestrians move along straight lines towards the target objective.

$$\begin{cases} \partial_t \rho + \nabla_x \cdot (\rho u) = 0 \\ \partial_t u + (u \cdot \nabla_x) u = \alpha(U(\rho)e_0 - u) - \frac{K^2(\rho)}{\rho} \nabla_{e_0} \rho \end{cases} \quad (2.1.15)$$

where  $\alpha > 0$  is the inverse of the relaxation time,  $e_0$  is a unit vector pointing towards the target and  $U(\rho)$  is an equilibrium speed of pedestrians. The second class of models refers to walkers that still move towards the target objective, but are also attracted by paths with small density gradients.

$$\begin{cases} \partial_t \rho + \nabla_x \cdot (\rho u) = 0 \\ \partial_t u + (u \cdot \nabla_x) u = \alpha(U(\rho)e - u) - \frac{K^2(\rho)}{\rho} \nabla_e \rho \end{cases} \quad (2.1.16)$$

where direction of motion is given by  $e = e_0 + e_1$ , with  $e_1$  being the correction term related to the attraction towards small density gradients.

The third class of models contains a pressure term which enables the momentum equation to predict the expected response of crowd behavior as time and space changes.

$$\begin{cases} \partial_t \rho + \nabla_x \cdot (\rho u) = 0 \\ \partial_t (u + P(\rho, u)u_0) + (u \cdot \nabla_x)(u + P(\rho, u)u_0) = \alpha(U(\rho)e_0 - u)\rho \end{cases} \quad (2.1.17)$$

where  $P = P(\rho, u)$  is some pressure that pedestrians feel along the preferred path, depending on pointwise crowding of the domain and on their current velocity.

### Gradient Flow Type

This approach is presented by Maury, Roundneff and Santambrogio [59]. The model is based on a strong expression of the congestion constraint. It rests on the following two principles.

- The principle population is described by a density  $\rho$  which is subjected to remain below a certain maximal value, this density follows an advection equation.
- The advecting field is the closet among the admissible fields, to some spontaneous field  $L$ , which corresponds to the strategy people would follow in the absence of others.

If we denote by  $C_\rho$  the cone of admissible velocities, the model takes the following form

$$\begin{cases} \partial_t \rho + \nabla \cdot (\rho u) = 0 \\ u = P_{C_\rho} L \end{cases} \quad (2.1.18)$$

where the projection is meant in the  $L^2$  sense. The spontaneous velocity field has a gradient structure  $L = -\nabla D$ , where  $D$  is the geodesic distance to the exit.

## 2.2 The Microscopic Model

We consider a microscopic social force model for pedestrian flow including an optimal path computation. It is developed by coupling the classical social

force models, see for example Ref. [40, 39], with a optimal path computation as in the Hughes approach to pedestrian flow, see Ref. [43]. We obtain a two-dimensional interacting particle system with locations  $x_i \in \mathbb{R}^2, i = 1, \dots, N$  and velocity  $v_i \in \mathbb{R}^2$ . The equations of motion are

$$\begin{aligned} dx_i &= v_i dt \\ dv_i &= \sum_{i \neq j} F(x_i - x_j, v_i - v_j) dt + G(x_i, v_i, \rho_i) dt - \frac{A^2}{2} v_i dt + A dW_t^{(i)}. \end{aligned}$$

The desired velocity and direction force is given by

$$G(x, v, \rho) = \frac{1}{T} \left( -U(\rho) \frac{\nabla \Phi(x)}{\|\nabla \Phi(x)\|} - v \right), \quad (2.2.1)$$

where

$$\rho = \rho(x) = \frac{1}{N_{max}^R} \sum_{j, \|x - x_j\| < R} 1$$

and  $\rho_i = \rho(x_i)$ . Here  $N_{max}^R$  is the maximal number of particles in a ball of radius  $R$  and  $T$  denotes the reaction time describing how fast pedestrians can correct their current velocity to the desired velocity. The function  $U : [0, 1] \rightarrow [0, U_{max}]$  describes the speed-density relationship.

The interaction force is given by

$$F(x, v) = F_{int}(x) + F_{diss}(x, v) \quad (2.2.2)$$

with the interaction force  $F_{int}$  given by

$$F_{int} = F_{int}(x) = -\nabla_x V(\|x\|)$$

where  $V$  is an interaction potential given by

$$V = V(x) = k_n \left( 2R^2 - \|x\| (2R - \frac{\|x\|}{2}) \right) H(2R - \|x\|)$$

where  $H$  is the Heaviside function. This yields

$$F_{int}(x) = k_n n(x) (2R - \|x\|) H(2R - \|x\|)$$

where

$$n = n(x) = \frac{x}{\|x\|}$$

is the normal unit vector. This force is complemented by a dissipative force, compare e.g. Ref. [40]. The dissipative force is given by

$$F_{diss} = (F_{diss}^n + F_{diss}^t) H(2R - \|x\|).$$

Here, the normal dissipative force is given by

$$F_{diss}^n(x, v) = -\gamma_n \langle v, n \rangle n.$$

The tangential friction force is

$$F_{diss}^t(x, v) = -\gamma_t v^t = -\gamma_t \langle v, n^\perp \rangle n^\perp$$

where

$$v^t = v - \langle v, n \rangle n$$

is the tangential unit vector pointing into the direction of the tangential component of the relative velocity and  $n^\perp$  is the normal to  $n$ .  $R$  denotes the radius of interaction of the pedestrians,  $k_n$  is the interaction constant and  $\gamma_n$  and  $\gamma_t$  are suitable positive friction constants.

Finally,  $W_t^{(i)}$  are independent Brownian motions in  $\mathbb{R}^2$  and  $A \geq 0$  is a constant.

### Remark 2.2.1 *Optimal Path*

Let  $\Phi$  be the travel costs for pedestrians to reach their destination. As one might expect, pedestrians intend to minimize these travel costs. Hughes [43, 44] proposed that pedestrians move in opposite to the gradient of the scalar potential  $\Phi$ , that is

$$-\frac{\nabla \Phi}{\|\nabla \Phi\|}. \quad (2.2.3)$$

The potential  $\Phi$  is determined by the nonlinear Eikonal equation

$$|\nabla \Phi| = g(\rho) \text{ in } \Omega, \quad (2.2.4)$$

$$\Phi = 0 \text{ on } \Omega_d \quad (2.2.5)$$

where  $\Omega_d$  is the destination for pedestrians and  $g(\rho)$  is a density-dependent cost function increasing in  $\rho$ . Pedestrians want to minimize the path length towards their destination but temper the estimated travel time by avoiding high densities. This behavior can be expressed by the 'density driven' rearrangement of the equipotential curves of  $\Phi$  using the cost function [43]

$$g(\rho) = \frac{1}{U(\rho)}.$$

**Remark 2.2.2** *A model with constant velocities which might be a more appropriate way to describe the actual movement of pedestrians is easily constructed in the following way. We define  $v = c\tau$  with  $\tau = (\cos(\alpha), \sin(\alpha)) \in S^1$  and assume that  $c > 0$  is constant. Then using  $d\tau = \tau^\perp d\alpha$  we obtain the equations*

$$\begin{aligned} dx_i &= c\tau_i dt \\ c d\alpha_i &= \sum_{i \neq j} \tau_i^\perp \cdot F(x_i - x_j, c(\tau_i - \tau_j)) dt - \frac{U(\rho_i)}{T \|\nabla \Phi(x_i)\|} \tau_i^\perp \cdot \nabla \Phi(x_i) dt + A dW_t^{(i)}, \end{aligned} \quad (2.2.6)$$

where  $W_t^{(i)}$  are Brownian motions in  $\mathbb{R}$ . Compare Refs. [22, 12] for constant velocity models in a biological context. In this model the direction of the pedestrians is changed according to the projection of the direction given by the gradient of the potential  $\Phi$  determined from the Eikonal equation and the projection of the interaction force onto the orthogonal direction of the motion. An individual based model with constant speed and the inclusion of a desired direction into an interaction potential can be found in Ref. [21].

**Remark 2.2.3** *Finite size effects with a minimal radius around a pedestrian could be included using interaction potentials with a singularity. Other variants are given by an elliptical interaction force*

$$f_{int}^n(x, v) = -\nabla_x V(x, v)$$

with a potential

$$V(x, v) = V(b(x, v))$$

with

$$b(x, v) = \|x\| + \|x - vT_e\|$$

or by a force including the human vision cone.

**Remark 2.2.4** *A local visibility model can be set up as follows. For each  $x \in \Omega$ ,  $\Phi$  is given by the solution of the Eikonal equation*

$$U(\rho(y)) \|\nabla \Phi\| - 1 = 0, y \in B_V(x)$$

where  $B_V(x)$  is a given circle around  $x$ . Boundary conditions on this circle are chosen for example as

$$\Phi(y) = \text{dist}(y, \Omega_D), y \in \partial B_V(x).$$

This would give a preference towards the desired direction. We refer to Ref. [5] for a local visibility model without using an Eikonal equation.

## 2.3 Mean Field Equation

Using the so-called 'weak coupling scaling' assumption [62, 9, 78] one rescales the interaction potential with the factor  $\frac{1}{N}$  where  $N$  denotes the total number of particles. Neglecting the stochastic force our scaled microscopic model states

$$\begin{aligned}\frac{dx_i}{dt} &= v_i \\ \frac{dv_i}{dt} &= \frac{1}{N} \sum_{i \neq j} F(x_i - x_j, v_i - v_j) + G(x_i, v_i, \rho_i).\end{aligned}\tag{2.3.1}$$

Letting  $N$  to infinity, one can derive in the limit of a large number of particles the associated mean field equation [78, 11, 10]. For completeness we briefly sketch the idea of the derivation. Let us denote by  $f^{(N)}(x_i, v_i, t)$  the  $N$ -particle probability density function, so that the probability of finding each of the  $i$  particles at position  $x_i$  and velocity  $v_i$  within a volume  $dx_i dv_i$  in phase space is  $f^{(N)}(x_i, v_i, t) \Pi_i dx_i dv_i$ . Conservation of mass allows to write the time evolution of  $f^{(N)}$  according to the Liouville equation

$$\partial_t f^{(N)} + \sum_{i=1}^N (\operatorname{div}_{x_i}(\dot{x}_i f^{(N)}) + \operatorname{div}_{v_i}(\dot{v}_i f^{(N)})) = 0 \tag{2.3.2}$$

The one-particle distribution function  $f^{(1)}(x_1, v_1, t)$  is defined as

$$f^{(1)}(x_1, v_1, t) = \int f^{(N)} dx_2 \cdots dx_N dv_2 \cdots dv_N. \tag{2.3.3}$$

Integrating Liouville eqn(2.3.2) over  $d\Omega_1 = dx_2 \cdots dx_N dv_2 \cdots dv_N$  one obtains

$$\partial_t f^{(1)} + \int \operatorname{div}_{x_1}(\dot{x}_1 f^{(N)}) d\Omega_1 + \int \operatorname{div}_{v_1}(\dot{v}_1 f^{(N)}) d\Omega_1 = 0 \tag{2.3.4}$$

The spatial divergence term reduces to  $v_1 \cdot \nabla_{x_1} f^{(1)}$ . For the velocity divergence term one obtains the two contributions

$$\operatorname{div}_{v_1} (G(x_1, v_1, \rho(x_1)) f^{(1)}) \tag{2.3.5}$$

with

$$\rho(x_1) := \int f^{(1)}(x_1, v) dv$$



and

$$\operatorname{div}_{v_1} \int \frac{1}{N} \sum_{j \neq 1} F(x_1 - x_j, v_1 - v_j) f^{(N)} d\Omega_1 \quad (2.3.6)$$

Since particles are indistinguishable we obtain for the last term

$$\operatorname{div}_{v_1} \frac{N-1}{N} \int F(x_1 - x_2, v_1 - v_2) f^{(2)} dx_2 dv_2 \quad (2.3.7)$$

with

$$f^{(2)} = \int f^{(N)} dx_3 \cdots dx_N dv_3 \cdots dv_N.$$

is the pair correlation function. Using a chaos assumption  $f^{(2)}(x_1, v_1, x_2, v_2) = f^{(1)}(x_1, v_1) f^{(1)}(x_2, v_2)$  we obtain finally the term

$$\operatorname{div}_{v_1} \frac{N-1}{N} \int F(x_1 - x_2, v_1 - v_2) f^{(1)}(x_2, v_2) dx_2 dv_2 f^{(1)}(x_1, v_1). \quad (2.3.8)$$

Thus, for  $N$  going to infinity one obtains for the distribution function  $f = f(x, v, t)$  of the particles the mean field equation

$$\partial_t f + v \cdot \nabla_x f + S f = 0 \quad (2.3.9)$$

with force term

$$S f = \nabla_v \cdot (G(x, v, \rho(x)) f(x, v)) + \nabla_v \cdot \left( \int \int F(x - y, v - w) f(y, w) dw dy f(x, v) \right)$$

where

$$\rho(x, t) := \int f(x, v, t) dv.$$

Adding the stochastic force gives an additional diffusion term on the right hand side, i.e.

$$\partial_t f + v \cdot \nabla_x f + S f = L f \quad (2.3.10)$$

with

$$L f = \frac{A^2}{2} \nabla_v \cdot (v f + \nabla_v f)$$

**Remark 2.3.1** *In case of the constant velocity model the mean field equation reads*

$$\partial_t f + c\tau \cdot \nabla_x f + S_\alpha f = L_\alpha f \quad (2.3.11)$$

*with force term*

$$\begin{aligned} S_\alpha f = & -\partial_\alpha \left( \frac{U(\rho(x))}{cT\|\nabla\Phi(x)\|} \tau^\perp \cdot \nabla\Phi(x) f \right) \\ & + \frac{1}{c} \partial_\alpha \left( \tau^\perp \cdot \int \int F(x-y, c(\tau-\tau')) f(y, \alpha') d\alpha' dy f \right) \end{aligned}$$

*and diffusion term*

$$L_\alpha f = \frac{A^2}{2c^2} \partial_{\alpha\alpha} f$$

For the following we define the momentum of  $f$  by

$$\rho u(x, t) := \int v f(x, v, t) dv.$$

Let us mention that the rigorous passage from microscopic particle systems towards the kinetic mean-field equation as  $N \rightarrow \infty$  has been treated for example in Ref. [10] or Ref. [78] for deterministic and stochastic case.

## 2.4 Hydrodynamic Models

Hydrodynamic limits for similar equations have been derived in Ref. [15, 11]. We consider the mean field equation

$$\partial_t f + v \cdot \nabla_x f + S f = L f$$

and derive different limit equations. Integrating against  $dv$  and  $v dv$  gives the continuity equation

$$\partial_t \rho + \nabla_x \cdot (\rho u) = 0 \quad (2.4.1)$$

and

$$\partial_t u + u \cdot \nabla_x u + \frac{1}{\rho} \nabla_x \int (v-u) \otimes (v-u) f(x, v) dv = \frac{1}{\rho} \int G(x, v) f(x, v) dv \quad (2.4.2)$$

$$+\frac{1}{\rho} \int \int \int F(x-y, v-w) f(y, w) dw dy f(x, v) dv - A^2 u$$

Considering an equation without diffusion, i.e. neglecting fluctuations and setting  $A = 0$ , it makes sense to use a moment closure approach with a mono-kinetic closure function

$$f \sim \rho(x) \delta_{u(x)}(v).$$

One obtains

$$\begin{aligned} & \int \int \int F(x-y, v-w) f(y, w) dw dy f(x, v) dv \\ &= \int \int \int F(x-y, v-w) \rho(y) \delta_{u(y)}(w) dw dy \rho(x) \delta_{u(x)}(v) dv \\ &= \rho(x) \hat{F}(\rho, u) \end{aligned}$$

with

$$\hat{F}(\rho, u) = \int F(x-y, u(x) - u(y)) \rho(y) dy.$$

Neglecting the velocity dependence of  $F$  one has

$$\hat{F}(\rho, u)(x) = \int F(x-y) \rho(y) dy = F \star \rho(x).$$

Alltogether

$$\partial_t u + (u \cdot \nabla_x) u = G(x, u, \rho) + \hat{F}(\rho, u). \quad (2.4.3)$$

with

$$G(x, u, \rho) = \frac{1}{T} \left( -U(\rho(x)) \frac{\nabla \Phi(x)}{\|\nabla \Phi(x)\|} - u \right)$$

This is coupled to

$$f(\rho(x)) \|\nabla \Phi(x)\| = 1.$$

Using other functions to close the equation with  $A \neq 0$  results in equations including a pressure term. One could for example use a closure with a standard Maxwellian as closure function leading to

$$\partial_t u + \frac{1}{\rho} \nabla_x \rho = G(x, u, \rho) + F \star \rho - A^2 u. \quad (2.4.4)$$

Compare this model to the second order models in Ref. [5].

**Remark 2.4.1** *In the constant velocity case the balance equations read after integrating the mean field equation*

$$\partial_t \int_0^{2\pi} f d\alpha + c \nabla_x \cdot \int_0^{2\pi} \tau f d\alpha = 0.$$

With  $\rho = \int_0^{2\pi} f d\alpha$  and  $\rho u = c \int_0^{2\pi} \tau f d\alpha$  this gives

$$\partial_t \rho + \nabla_x \cdot (\rho u) = 0 \quad (2.4.5)$$

as before. Multiplying the mean field equation with  $c\tau$  and integrating one obtains

$$\begin{aligned} & \partial_t(\rho u) + \nabla_x \cdot P + \frac{U(\rho)}{T \|\nabla \Phi\|} \int \tau^\perp \nabla \Phi \cdot \tau^\perp f d\alpha \\ & - \int \tau^\perp \left( \int \int F(x-y, c(\tau-\tau')) f(y, \alpha') d\alpha' dy \right) \cdot \tau^\perp f d\alpha = -\frac{A^2}{2c^2} \rho u \end{aligned}$$

with

$$P = c^2 \int \tau \otimes \tau f d\alpha.$$

Assuming that  $F$  does not depend on the velocity and using  $\tau^\perp \otimes \tau^\perp = I - \tau \otimes \tau$  this simplifies to

$$\partial_t(\rho u) + \nabla_x \cdot P + \left( \rho I - \frac{1}{c^2} P \right) \left( \frac{U(\rho)}{T \|\nabla \Phi\|} \nabla \Phi - F \star \rho \right) = -\frac{A^2}{2c^2} \rho u. \quad (2.4.6)$$

Considering again the case  $A = 0$  and using the monokinetic closure  $\rho \delta_u(\tau)$  one obtains

$$\partial_t u + (u \cdot \nabla_x) u + \left( I - \frac{1}{c^2} u \otimes u \right) \left( \frac{U(\rho)}{T \|\nabla \Phi\|} \nabla \Phi - F \star \rho \right) = 0. \quad (2.4.7)$$

Using for the case  $A \neq 0$  a constant function  $\frac{\rho}{2\pi}$  as closure gives

$$\partial_t u + \frac{1}{2\rho} \nabla_x \rho + \frac{1}{2} \left( \frac{U(\rho)}{T \|\nabla \Phi\|} \nabla \Phi - F \star \rho \right) = -\frac{A^2}{2c^2} u. \quad (2.4.8)$$

In the constant velocity case there are other classical choices for a closure function if  $A \neq 0$ , for example, a so called maximum entropy closure using as closure function the function

$$f = a \exp(b \cdot \tau)$$

where  $a$  and  $b$  are determined from  $\rho$  and  $u$  by the moment conditions on  $f$ . We refer to Refs. [19, 57] for the classical case of radiative transfer. We note that this distribution is sometimes called the van Mises Fisher distribution. This yields

$$P = \rho D(u) \quad (2.4.9)$$

with

$$\rho = \langle a \exp(b \cdot \tau) \rangle \quad (2.4.10)$$

$$\rho u = c \langle a \tau \exp(b \cdot \tau) \rangle \quad (2.4.11)$$

$$u = u(b) = c \frac{\langle \tau \exp(b \cdot \tau) \rangle}{\langle \exp(b \cdot \tau) \rangle} \quad (2.4.12)$$

$$D = D(b) = c^2 \frac{\langle \tau \otimes \tau \exp(b \cdot \tau) \rangle}{\langle \exp(b \cdot \tau) \rangle} \quad (2.4.13)$$

$D$  and  $u$  can be written explicitly as functions of  $b$ :

$$u = c \frac{I_1(|b|)}{|b| I_0(|b|)} b \quad (2.4.14)$$

$$|u| = c \frac{I_1(|b|)}{I_0(|b|)} \quad (2.4.15)$$

$$D = c^2(1 - \chi) \text{Id} + c^2 \frac{(2\chi - 1)}{|b|^2} b \otimes b \quad (2.4.16)$$

$$\chi = \frac{1}{2\pi\rho} \int \left( \tau \cdot \frac{b}{|b|} \right)^2 a \exp(b \cdot \tau) d\alpha = \frac{1}{2} \left( 1 + \frac{I_2(|b|)}{I_0(|b|)} \right), \quad (2.4.17)$$

where  $I_\nu$  is again the  $\nu$ -th modified Bessel function of the first kind. One inverts the relation between  $|u|$  and  $|b|$  which can be proven to define a bijection. Then, one uses  $|b|(|u|)$  in the definition of  $\chi(|b|)$  to obtain  $D(|b|(u))$ . Together one obtains the model

$$\partial_t \rho + \nabla_x \cdot (\rho u) = 0 \quad (2.4.18)$$

$$\partial_t(\rho u) + \nabla_x \cdot (D(u)\rho) + \left( I - \frac{1}{c^2} D(u) \right) \left( \frac{U(\rho)}{T \|\nabla \Phi\|} \nabla \Phi - F \star \rho \right) \rho = -\frac{A^2}{2c^2} \rho u$$

For a discussions of similar closures for other pedestrian models we refer to Ref. [21]. Moreover, we note that for small  $u$  one has  $D(u) \sim \frac{c^2}{2} I$ . Using this in (2.4.18) one obtains again equation (2.4.8).

## 2.5 Scalar Models

In this section, we reduce the hydrodynamic description deriving scalar models. We assume again an interaction potential depending only on  $x$  and neglect the dissipative forces. Starting from the hydrodynamic momentum equation derived from the standard Maxwellian closure we neglect time changes in this equation and obtain an equation for  $u$ . Solving for  $u$  the result is used to close the continuity equation. This procedure gives

$$\frac{1}{\rho} \nabla_x \rho + U(\rho) \frac{\nabla \Phi(x)}{\|\nabla \Phi(x)\|} - T F \star \rho = -(1 + A^2 T) u$$

The resulting scalar equation for  $\rho$  is after rescaling the time with  $1/(1+A^2T)$

$$\partial_t \rho - \nabla_x \left( U(\rho(x)) \frac{\nabla \Phi(x)}{\|\nabla \Phi(x)\|} \rho \right) + \nabla_x (T(F \star \rho) \rho) = T \Delta_x \rho. \quad (2.5.1)$$

**Remark 2.5.1** *For the constant velocity model the same procedure gives for the hydrodynamic model derived from the mono kinetic closure a trivial velocity. We start from the hydrodynamic model with constant closure function and obtain*

$$\frac{1}{\rho} \nabla_x \rho + \frac{1}{2} \left( \frac{U(\rho)}{T \|\nabla \Phi\|} \nabla \Phi - F \star \rho \right) = -\frac{A^2}{c^2} u \quad (2.5.2)$$

The resulting scalar equation is after rescaling the time with  $c^2/(TA^2)$

$$\partial_t \rho - \nabla_x \left( U(\rho(x)) \frac{\nabla \Phi(x)}{\|\nabla \Phi(x)\|} \rho \right) + \nabla_x (T(F \star \rho) \rho) = c^2 T \Delta_x \rho \quad (2.5.3)$$

that means the same equation as before up to a scaling of the diffusion term.

**Remark 2.5.2** *To derive an associated diffusive equation we write the force  $F$  as a gradient field with  $F = -\nabla V$ . Assuming that we are in a dense situation we approximate the potential  $V$  by*

$$V(y) \sim D \delta_0(y)$$

with the constant  $D > 0$  given by

$$D = \int V(y) dy.$$

*The symmetry of the convolution*

$$\int \nabla_x V(x-y) \rho(y) dy = - \int \nabla_y V(x-y) \rho(y) dy = \int V(x-y) \nabla_y \rho dy$$

*and the above localization gives*

$$F \star \rho = - \int V(x-y) \nabla_y \rho dy \sim -D \nabla_x \rho$$

*and therefore neglecting inertia effects*

$$\partial_t \rho - \nabla_x (U(\rho(x)) \frac{\nabla \Phi(x)}{\|\nabla \Phi(x)\|} \rho) = DT \nabla_x (\rho \nabla \rho) \quad (2.5.4)$$

*This is combined with the Eikonal equation*

$$U(\rho(x)) \|\nabla \Phi(x)\| = 1.$$

*Thus, we have obtained a diffusive version of the Hughes equation, see Ref. [43].*

**Remark 2.5.3** *We note that (2.5.3) is similar to an equation considered in Refs. [16, 17]. There the equation*

$$\partial_t \rho - \nabla_x \left( U(\rho(x)) \frac{\nabla \Phi(x)}{\|\nabla \Phi(x)\|} \rho \right) + \epsilon \nabla_x \left( U(\rho) \frac{\nabla \eta \star \rho}{\sqrt{1 + \|\nabla \eta \star \rho\|^2}} \rho \right) = 0 \quad (2.5.5)$$

*has been considered, where  $\Phi$  is the solution of the homogeneous Eikonal equation and  $\eta$  is a mollifier. This is equivalent to (2.5.3) if  $\eta$  is identified with  $V$  and*

$$T(\rho) = \frac{\epsilon U(\rho)}{\sqrt{1 + \|\nabla \eta \star \rho\|^2}}.$$





# Chapter 3

## The Eikonal Equation

This chapter is devoted for the detailed discussion of the Eikonal equation. This chapter is organized as follows. We start with a brief introduction about the Eikonal equation followed by summarizing the existing numerical methods to solve this equation. After that we present a mesh free method to solve the Eikonal equation. Finally we discuss few numerical examples in order to validate our method.

### 3.1 Introduction

The Eikonal equation

$$\|\nabla\Phi(x)\| = F(x), \quad x \in \Omega \quad F(x) > 0 \quad (3.1.1)$$

subject to the boundary condition

$$\Phi = g(x), \quad x \in \Gamma$$

is a non linear first order hyperbolic partial differential equation, where  $\Omega$  is a domain in  $\mathbb{R}^2$ ,  $\Phi$  the Eikonal,  $F(x)$  the slowness field and  $g(x)$  a Lipschitz continuous function and is given.

The Eikonal equation is of significant interest in the field of numerical analysis. This equation has got many applications in differential games [45], image processing and computer vision [42, 64, 72], optimal control [3], robotic path planning [84], shape from shading [49] and segmentation [27].

We may apply the classical method of characteristics to solve the Eikonal equation in phase space. Although the characteristics may never intersect

in phase space, their projection into physical space may intersect so that the solution in physical space is not uniquely defined at these intersections. Crandall and Lions [20] introduced the concept of viscosity solutions for Hamilton-Jacobi equations so that a unique weak solution can be defined for such first order non linear equations.

The main aspect of any numerical algorithm for the Eikonal equation is the derivation of consistent and accurate discretization scheme. The method has to satisfy the causality condition of the Eikonal equation and has to deal the non-differentiability at intersections of characteristics properly. The other main aspect of a numerical method is that it has to be efficient to solve a large system of non linear equations.

There are mainly two approaches for solving the Eikonal equation. The first approach of numerical methods is based on reformulating the equation into suitable time-dependent problem. Osher[66] provides a natural link between static and time-dependent Hamilton-Jacobi equations by using the level set idea and thus raising the problem one-dimension higher. However many time steps may be needed for the convergence of the solution in the entire domain due to finite speed of propagation and CFL condition for stability. The other approach is to treat the problem as a stationary boundary value problem and design an efficient numerical algorithm to solve the system of non linear equations after discretization.

Several methods have been proposed to solve the Eikonal equation [1, 85, 68, 48, 30, 73, 74, 92, 46, 69]. The most stable methods among those are the fast marching method and the fast sweeping method. The fast sweeping method [92, 47, 87] is an iterative algorithm with optimal complexity that finds the numerical solution by using the non linear upwind method and Gauss-seidel type iterations with alternating sweepings in pre-determined directions. The fast marching method [1, 73, 50, 74] combines entropy-satisfying upwind schemes and a fast sorting technique to find the solution in one-pass algorithm.

A recent computational study [33] of the fast marching and fast sweeping methods for solving the Eikonal equation has shown that on realistic grids, the fast sweeping method is faster than the fast marching method for problems with simple geometry. However, on a fixed grid, for non-uniform problems and for complex geometry, the fast marching method is faster.

In the fast marching method, the update of the solution follows the causality in a sequential way i.e., the solution is updated one grid point by one grid point in the order that the solution is strictly increasing. Hence an upwind difference scheme and a heap sort algorithm is needed. The complexity is of

order  $O(N \log N)$  for  $N$  grid points, where the  $\log N$  factor comes from the heap sort algorithm. The fast marching methods are applicable to cartesian domains [1, 73] and triangulated surfaces [50, 74]. But it is not applicable to the domain, where the grid points are randomly distributed. Also, in case of triangulated surfaces, if the triangles are not acute, then the fast marching method is complicated to use. We present a mesh free method, which is simple to implement and applicable to any arbitrary grid distribution and complex geometries.

We used the least square approximation to approximate the derivatives in the Eikonal equation as in the finite point-set method [81, 82] and we combine it with the idea of front marching in the fast marching method. The time complexity of the method is of order  $O(N \log N)$  for  $N$  grid points same as in the fast marching method. The advantage of our method over the fast marching method is that it is simple to use and it will work with any arbitrary grid structure.

## 3.2 Existing Methods

Many numerical methods are available to solve the Eikonal equation, but fast marching method and fast sweeping method are the popular methods among them. We present a brief overview of these methods in this section.

### 3.2.1 Upwind Approximation of the Eikonal Equation

Consider the finite difference approximation introduced in [65] namely

$$\Phi_x^2 = (\max(D_i^{-x}\Phi, 0)^2 + \min(D_i^{+x}\Phi, 0)^2) \quad (3.2.1)$$

where  $D_i^{+x}$  and  $D_i^{-x}$  are standard finite difference notations that

$$D_i^{-x} = \frac{\Phi_i - \Phi_{i-1}}{h} \quad (3.2.2)$$

and

$$D_i^{+x} = \frac{\Phi_{i+1} - \Phi_i}{h} \quad (3.2.3)$$

Here  $\Phi_i$  is the value of  $\Phi$  on a grid at the point  $ih$  with grid spacing  $h$ . Eqn (3.2.1) is an upwind scheme; it chooses grid points in the approximation in terms of the direction of the flow of information. Extending this idea

of upwind approximation for the gradient to two dimensions, we have the scheme for the Eikonal equation,

$$[(\max(D_{ij}^{-x}\Phi, 0)^2 + \min(D_{ij}^{+x}\Phi, 0)^2) + (\max(D_{ij}^{-y}\Phi, 0)^2 + \min(D_{ij}^{+y}\Phi, 0)^2)]^{1/2} = F_{ij}$$

The forward and backward operators  $D^{-y}$  and  $D^{+y}$  in the  $y$  coordinate direction are similar to the one defined for the  $x$  direction in (3.2.2) and (3.2.3). A slightly different upwind scheme [71], which will turn out to be more convenient is given by

$$[\max(\Phi_{ij}^{-x}, -\Phi_{ij}^{+x}, 0)^2 + \max(\Phi_{ij}^{-y}, -\Phi_{ij}^{+y}, 0)^2]^{1/2} = F_{ij} \quad (3.2.4)$$

To solve the Eqn (3.2.4), one approach is through iteration[71]. Consider a stencil of a grid point and its four neighbors, as shown in Fig 3.1. Eqn (3.2.4) is a piece wise quadratic equation for  $\Phi_{ij}$ , assuming that the neighbouring grid values for  $\Phi$  are given. If we assume  $N$  points in the domain and this method takes roughly  $N$  steps to converge, then the operation count for this method is  $O(N^2)$ , which is expensive.

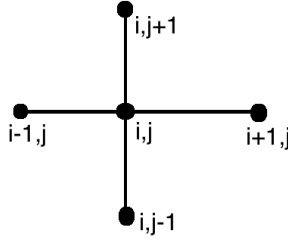


Figure 3.1: Grid stencil.

### 3.2.2 Fast Marching Method

Fast marching method is based on two key components. First by exploiting upwind viscosity schemes, it automatically select solutions which include non-differentiability in natural ways. Second, by coupling the causality of

these schemes to fast sorting methods, it becomes extremely efficient computationally. The complexity of fast marching method algorithm is  $O(N \log N)$ , where  $N$  is the total number of points in the domain  $\Omega$ .

The central idea behind the Fast marching method is to systematically construct the solution in a "downwind" fashion to produce the solution and to reduce the time complexity. The upwind difference structure of Eqn (3.2.4) means that the information propagates one way, that is from smaller values to larger values of  $\Phi$ . Hence, the Fast marching algorithm rests on solving Eqn (3.2.4) by building the solution outwards from the smallest value of  $\Phi$ . The algorithm is made fast by confining the "building zone" to a narrow band around the front. This approach is motivated by the narrow band technology [14] used in recovering shapes in images. The idea is to sweep the front ahead in a downward fashion by considering a set of points in narrow band around the existing front and to march this narrow band forward, freezing the values of existing points and bringing new ones into the narrow band structure. The key is in the selection of which grid point in the narrow band to update first.

Consider the problem to solve the Eikonal equation where the boundary value is known at the origin. It is represented in Fig 3.2 where the black sphere at  $(0,0)$  represents a grid point where the value of  $\Phi$  is known and the light grey spheres are grid points where the solution is unknown. Fast

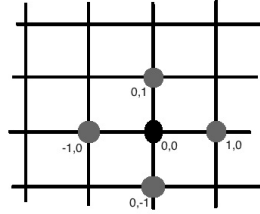


Figure 3.2: Beginning of the Fast marching method.

marching method algorithm starts by marching "downward" from the known value, computing new values at each of four neighboring grid points. This provides possible values for  $\Phi$  at each grid point  $(-1,0)$ ,  $(1,0)$ ,  $(0,-1)$  and  $(0,1)$ . To proceed further, the question is which one of these four grid points to choose. The answer lies in the observation that the smallest  $\Phi$  value at

these four grids must be correct. Because of upwinding, no point can be affected by grid points containing larger values of  $\Phi$ . Thus one can freeze the value of  $\Phi$  at this smallest grid point and proceed ahead with the algorithm.

Another way to look at this technique is that each minimum trial value begins an application of Huygens' principle and the expanding wave front touches and updates all others. The speed of the algorithm comes from a heap sort technique to efficiently locate the smallest element in narrow band.

The Fast marching method algorithm is as follows: First, tag points in the initial conditions as *Accepted*. Then tag *Narrow band* for all points one grid point away from *Accepted*. Finally all of the remaining grid points are tagged as *Far away*. Then the loop is

- Begin loop: Let *Trial* be the point in *Narrow band* with the smallest value of  $\Phi$ .
- Tag as *Narrow band* all neighbors of *Trial* that are not *Accepted*. If the neighbor is in *Far away*, remove it from that list and add it to the set *Narrow band*.
- Recompute the values of  $\Phi$  at all *Narrow band* neighbors of *Trial* by solving the piecewise quadratic equation according to eqn (3.2.4).
- Add the point *Trial* to *Accepted*, remove it from *Narrow band*.
- Return to top of the loop.

### Heap sort and computational efficiency

The key to an efficient version of the above technique lies in the fast way of locating grid point in the narrow band with the smallest value for  $\Phi$ . There are several ways to store the *Trial* elements so that one can easily find the smallest element. In Fast marching method, suppose that there is a ordered structure of the elements in *Trial*. When a point is updated, its neighbors get updated and their  $\Phi$  values may change. Thus only a small subset of the structure must be re-ordered in order to regain the ordering.

This leads quite naturally to a variation on a heap algorithm with back pointers to store the  $\Phi$  values. Specifically fast marching method use a min-heap data structure, see Remark 3.2.1 . The values of  $\Phi$  are stored, together with the indices which give their location in the grid structure. The marching algorithm works by first looking for the smallest element in the *Narrow band*, this operation involves deleting the root and one sweep of *Downheap*

to ensure that the remaining elements satisfy the heap property. The algorithm proceeds by tagging the neighboring points that are not *Accepted*. The *Far away* neighbors are added to the heap using an **Insert** operation and values at the remaining points are updated using Eqn (3.2.4). **Insert** works by increasing the heap size by one and trickling the new element upward to its correct location using an **Upheap** operation. lastly, to ensure that the updated  $\Phi$  values do not violate the heap property, we need to perform an **Upheap** operation starting at the location and proceeding up to the tree. The **Downheap** and **Upheap** operations carry an element all the way from root to bottom or vice versa in worst case. Therefore, this takes  $(O(\log M))$  time assuming there are  $M$  elements in the heap. Note that the heap, which is a complete binary tree, is always guaranteed to remain balanced. All that remains is the operation of searching for the *Narrow band* neighbors of the smallest element in the heap. This can be done in  $O(1)$  by maintaining back pointers from the grid to the heap array.

Since the total work in changing the value of one element of the heap and bubbling its value upwards is  $O(\log M)$ , where  $M$  is the size of the heap and  $M$  is bounded by  $N$ , this produces a total operation count of  $N \log N$  for the fast marching method on a grid of  $N$  points. Thus, if there are  $N$  points in total domain, the Fast marching method reduces the total operation count from  $N^2$  to  $N \log N$ . Essentially, each grid point is visited once to compute the solution.

**Remark 3.2.1** *A min-heap is a complete binary tree with a property that the value at any given node is less than or equal to the values at its children. It is more efficient to represent a heap sequentially as an array by storing a node at location  $k$  and its children at locations  $2k$  and  $2k + 1$ . From the definition, the parent of a given node at  $k$  is located at  $k/2$ . Therefore, the root which contains the smallest element is stored at locations  $k = 1$  in the array. Finding the parent or children of a given element are simple array accesses which take  $O(1)$  time. The **Upheap** operation is used to add a node to a heap. When you upheap a node, you compare its value to its parent node; if its value is less than its parent node, then you switch the two nodes and continue the process. Otherwise the condition is met that the parent node is less than the child node and so you can stop the process. Once you find a parent node that is less than the node being upheaped, you know that the heap is correct if the node being upheaped is greater than its parent, and its parent is greater than its own parent, all the way up to the root. The **Downheap** process is similar to the upheaping process. When you downheap a node, you compare its value with its two children. If the node is less than both of its children, it remains in place; otherwise, if it is greater than one or both of*

*its children, then we switch it with the child of lowest value, thereby ensuring that, of the three nodes being compared, the new parent node is lowest. The node being downheaped is in its proper position, it may be greater than one or both of its new children; the downheap process must be repeated until the node is less than both of its children.*

### 3.2.3 Fast Sweeping Method

Consider the Eikonal equation (3.1.1) with boundary condition  $\Phi(x) = 0$  for  $x \in \Gamma$ . Let  $m$  and  $n$  be the number of grid points in  $x$  and  $y$  direction respectively. Fast sweeping method aims at improving on the Gauss-Jacobi method by using a Gauss-Seidel type update process. It avoids the ordering step of fast marching method, instead it considers sweeps in predetermined directions. Fast sweeping method also uses the upwind difference scheme [71] to discretize the Eikonal equation, given in eqn (3.2.4) at the interior grid points to enforce the causality. One sided difference is used at the boundary of the computational domain.

To enforce the boundary condition  $\Phi(x) = 0$  for  $x \in \Gamma$ , assign exact values or interpolated values at grid points in or near  $\Gamma$ . These values are fixed in later calculations. For all other grid points assign large positive values. Fast sweeping method proceeds with Gauss-Seidel iterations with alternative sweeping orders.

At each grid point  $x_{ij}$  whose value is not fixed during the initialization, compute the solution denoted by  $\bar{\Phi}$  from the current values of its neighbors using the eqn (3.2.4) and then update  $\Phi_{ij}$  to be the smaller one between  $\bar{\Phi}$  and its current value. i.e.,

$$\Phi_{ij}^{new} = \min(\Phi_{ij}^{old}, \bar{\Phi}) \quad (3.2.5)$$

Sweep the whole domain with four alternating orderings repeatedly:

1.  $i=1: m, j=1:n$  (upper left to lower right)
2.  $i=m:1, j=1: n$  (lower left to upper right)
3.  $i=m:1, j=n:1$  (lower right to upper left)
4.  $i=1:m, j=n:1$  (upper right to lower left)

Stopping criteria for the fast sweeping method is for given  $\delta > 0$ ,  $|\Phi_{ij}^{new} - \Phi_{ij}^{old}|_{\infty} < \delta$ . If all grid points can be ordered according to the causality along characteristics, one iteration of the Gauss-Seidel iteration is enough



for convergence. The key point behind Gauss-Seidel iterations with different sweeping ordering is that each sweep will follow the causality of a group of characteristics in certain directions simultaneously and all characteristics can be divided into a finite number of such groups according to their directions. The value at each grid point is always non increasing during the iterations due to the updating rule. Whenever a grid point obtains the minimal value it can reach, the value is the correct value and the value will not be changed in later iterations.

Since fast sweeping method avoids the ordering process of grid points, the complexity of the algorithm is of order  $O(N)$ . However, in case of complex geometries or for non uniform problems it is computationally expensive then the fast marching method [33].

### 3.3 Finite Pointset Method

In this section, we present a mesh free method to solve the Eikonal equation. Let  $N$  be the number of nodes in the domain  $\Omega$ . Let  $A$  be the set of nodes, where the solution is already computed and  $B$  be the set of nodes where the solution yet to compute, see Fig 3.3

Let  $\{X_1, X_2, \dots, X_N\}$  be arbitrarily distributed points in  $\Omega$ . Let  $\tilde{X}$  be the node in  $B$ , where the solution is to be computed. For a given radius  $h$ , assume  $N_x$  is the set of neighbours of  $\tilde{X}$  and  $N_{xa}$  is the intersection of  $N_x$  and  $A$ .

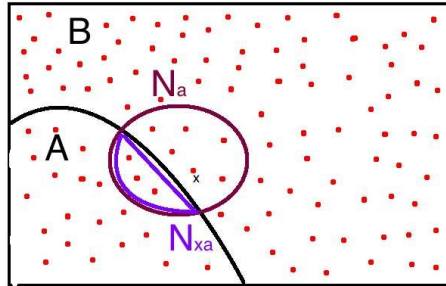


Figure 3.3: Domain.

We will now approximate the solution at  $\tilde{X}$  for three cases.

### 3.3.1 If cardinality of the set $N_{xa}$ is 1

If  $N_{xa} = \{X_1\}$ , then we compute the solution at ' $\tilde{X}$ ' directly by

$$\Phi_{\tilde{X}} = \Phi_{X_1} + F(\tilde{X})d(\tilde{X}, X_1) \quad (3.3.1)$$

where  $d(\tilde{X}, X_1)$  is the distance from  $\tilde{X}$  to  $X_1$ .

### 3.3.2 If cardinality of the set $N_{xa}$ is 2

Assume  $N_{xa} = \{X_1, X_2\}$ . Let  $\Phi_1$  and  $\Phi_2$  be the solutions at  $X_1$  and  $X_2$  respectively.

Consider the Taylor series expansion of  $\Phi$  around  $\tilde{X}$  with respect to  $X_1$  and  $X_2$ ,

$$\Phi_i = \Phi + dx_i\Phi_x + dy_i\Phi_y + e_i, i = 1, 2 \quad (3.3.2)$$

where  $e_i$  is the truncation error in the Taylor series expansion.

Expansion of the equation 3.3.2 gives,

$$\begin{aligned} \Phi_1 &= \Phi + dx_1\Phi_x + dy_1\Phi_y \\ \Phi_2 &= \Phi + dx_2\Phi_x + dy_2\Phi_y \end{aligned} \quad (3.3.3)$$

Equation (3.3.3) can be written as,

$$\begin{aligned} \Phi_1 - \Phi &= dx_1\Phi_x + dy_1\Phi_y \\ \Phi_2 - \Phi &= dx_2\Phi_x + dy_2\Phi_y \end{aligned} \quad (3.3.4)$$

$$M.a = b \quad (3.3.5)$$

where

$$M = \begin{pmatrix} dx_1 & dy_1 \\ dx_2 & dy_2 \end{pmatrix}; b = \begin{pmatrix} \Phi_1 - \Phi \\ \Phi_2 - \Phi \end{pmatrix}; a = \begin{pmatrix} \Phi_x \\ \Phi_y \end{pmatrix}$$

By letting  $Q = M^{-1}$ ,

$$\begin{pmatrix} \Phi_x \\ \Phi_y \end{pmatrix} = Q_{2 \times 2} \cdot \begin{pmatrix} \Phi_1 - \Phi \\ \Phi_2 - \Phi \end{pmatrix} \quad (3.3.6)$$

$$\begin{pmatrix} \Phi_x \\ \Phi_y \end{pmatrix} = \begin{pmatrix} q_{11} & q_{12} \\ q_{21} & q_{22} \end{pmatrix} \begin{pmatrix} \Phi_1 - \Phi \\ \Phi_2 - \Phi \end{pmatrix} \quad (3.3.7)$$

$$\begin{pmatrix} \Phi_x \\ \Phi_y \end{pmatrix} = \begin{pmatrix} q_{11}\Phi_1 + q_{12}\Phi_2 - \Phi(q_{11} + q_{12}) \\ q_{21}\Phi_1 + q_{22}\Phi_2 - \Phi(q_{21} + q_{22}) \end{pmatrix} \quad (3.3.8)$$

By comparing both sides row wise, we have

$$\Phi_x = q_{11}\Phi_1 + q_{12}\Phi_2 - \Phi(q_{11} + q_{12}) \quad (3.3.9)$$

and

$$\Phi_y = q_{21}\Phi_1 + q_{22}\Phi_2 - \Phi(q_{21} + q_{22}) \quad (3.3.10)$$

Now consider the equation (3.3.9)

$$\Phi_x = q_{11}\Phi_1 + q_{12}\Phi_2 - \Phi(q_{11} + q_{12}) \quad (3.3.11)$$

This can be written as

$$\Phi_x = a_1 + \Phi a_2 \quad (3.3.12)$$

with

$$a_1 = q_{11}\Phi_1 + q_{12}\Phi_2$$

and

$$a_2 = -(q_{11} + q_{12})$$

Similarly from the equation (3.3.10), we get

$$\Phi_y = b_1 + \Phi b_2 \quad (3.3.13)$$

with

$$b_1 = q_{21}\Phi_1 + q_{22}\Phi_2$$

and

$$b_2 = -(q_{21} + q_{22})$$

From the Eikonal equation (3.1.1), we have

$$\Phi_x^2 + \Phi_y^2 = F^2. \quad (3.3.14)$$

If we substitute (3.3.12) and (3.3.13) in (3.3.14), we get

$$(a_2^2 + b_2^2)\Phi^2 + 2(a_1a_2 + b_1b_2)\Phi + a_1^2 + b_1^2 = F^2 \quad (3.3.15)$$

which is a quadratic equation in  $\Phi$ , which yields two solutions.

The problem here is to determine which of these two solutions do we take as the appropriate one. The upwind difference structure of the Eikonal equation means that the information propagates from smaller values to larger values of  $\Phi$ , i.e, we take as the computed value of  $\Phi$  the larger. We also need to check whether the approximated solution at  $\tilde{X}$  satisfies causality condition: the characteristic passing through  $\tilde{X}$  is in between the two vectors  $\overrightarrow{\tilde{X}X_1}$  and  $\overrightarrow{\tilde{X}X_2}$ . This is a crucial condition for the monotonicity of the scheme.

### 3.3.3 If cardinality of the set $N_{xa}$ is greater than 2

Assume  $N_{xa} = \{X_1, X_2, \dots, X_n\}$ . If there are more than two elements in  $N_{xa}$ , then we use least square approximation to approximate the solution at  $\tilde{X}$ .

First, we sort all the elements from  $N_{xa}$  with respect to the angles from  $e_1 = (1, 0)$  to  $\tilde{X}X_i$ . Let  $N_{xa}^{up}$  be a subset of  $N_{xa}$ , which consists of elements from  $N_{xa}$  with angles between 0 and 180 degrees in the ascending order. Let  $N_{xa}^{down}$  be a subset of  $N_{xa}$ , which consists of elements from  $N_{xa}$  with angles between 180 and 360 degrees in the ascending order. Let  $k_1$  and  $k_2$  be the number of elements in  $N_{xa}^{up}$  and  $N_{xa}^{down}$ , respectively.

We approximate the solution at  $\tilde{X}$  in four steps.

- In first step, we approximate the solution by considering two end points of  $N_{xa}^{up}$  by the method explained in subsection 3.3.2. If the computed solution follows the causality condition, then we approximate the solution by using all elements from  $N_{xa}^{up}$  by the least square approximation explained below.
- Next, we approximate the solution by considering the last element of  $N_{xa}^{up}$  and the first element of  $N_{xa}^{down}$ .
- In third step, we approximate the solution by considering two end points of  $N_{xa}^{down}$  and if it follows causality condition, then we compute the solution using all the elements from  $N_{xa}^{down}$ .
- In final step, we consider the last element from  $N_{xa}^{down}$  and the first element from  $N_{xa}^{up}$  and approximate the solution.

Depending on the size of  $N_{xa}^{up}$  and  $N_{xa}^{down}$ , we skip some steps in the above procedure. If there are multiple possible solutions, we chose the minimum among them.

#### Least square approximation of the Eikonal equation:

Suppose we have to approximate the solution at  $\tilde{X}$  and  $\{X_1, X_2, \dots, X_n\}$  is the set of neighbours of  $\tilde{X}$  within the radius of  $h$ .

Consider the Taylor series expansion of  $\Phi$  around  $\tilde{X}$

$$\Phi(X_i) = \Phi + dx_i \Phi_x + dy_i \Phi_y + e_i, \quad i = 1, 2, \dots, n, \quad (3.3.16)$$

where  $e_i$  is the truncation error in the Taylor series expansion.

Expansion of the equation (3.3.16) gives,

$$\begin{aligned} \Phi_1 - \Phi &= dx_1 \Phi_x + dy_1 \Phi_y \\ \Phi_2 - \Phi &= dx_2 \Phi_x + dy_2 \Phi_y \\ &\vdots \\ \Phi_n - \Phi &= dx_n \Phi_x + dy_n \Phi_y \end{aligned} \quad (3.3.17)$$

If we write the equation (3.3.17) in matrix form,

$$M.a = b, \quad (3.3.18)$$

where

$$M = \begin{pmatrix} dx_1 & dy_1 \\ dx_2 & dy_2 \\ \vdots & \vdots \\ dx_n & dy_n \end{pmatrix}; b = \begin{pmatrix} \Phi_1 - \Phi \\ \Phi_2 - \Phi \\ \vdots \\ \Phi_n - \Phi \end{pmatrix}; a = \begin{pmatrix} \Phi_x \\ \Phi_y \end{pmatrix}.$$

Now, multiply both sides of the equation (3.3.18) with  $M^T W$ , where

$$W = \begin{pmatrix} w_1 & \cdot & \cdot & 0 \\ 0 & w_2 & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot \\ 0 & \cdot & \cdot & w_n \end{pmatrix}$$

is the weight matrix with Gaussian weight  $w_i = \exp^{-6.25 \frac{-\|\tilde{X} - X_i\|^2}{h^2}}$ . Then equation (3.3.18) is equivalent to

$$\begin{pmatrix} \sum_i w_i dx_i^2 & \sum_i w_i dx_i dy_i \\ \sum_i w_i dx_i dy_i & \sum_i w_i dy_i^2 \end{pmatrix} \begin{pmatrix} \Phi_x \\ \Phi_y \end{pmatrix} = \begin{pmatrix} \sum_i w_i dx_i (\Phi_i - \Phi) \\ \sum_i w_i dy_i (\Phi_i - \Phi) \end{pmatrix} \quad (3.3.19)$$

This can be written as,

$$P_{2 \times 2} \begin{pmatrix} \Phi_x \\ \Phi_y \end{pmatrix} = \begin{pmatrix} \sum_i w_i dx_i (\Phi_i - \Phi) \\ \sum_i w_i dy_i (\Phi_i - \Phi) \end{pmatrix}$$

with

$$P_{2 \times 2} = \begin{pmatrix} \sum_i w_i dx_i^2 & \sum_i w_i dx_i dy_i \\ \sum_i w_i dx_i dy_i & \sum_i w_i dy_i^2 \end{pmatrix}. \quad (3.3.20)$$

By letting  $Q = P^{-1}$ ,

$$\begin{pmatrix} \Phi_x \\ \Phi_y \end{pmatrix} = Q \cdot \begin{pmatrix} \sum_i w_i dx_i (\Phi_i - \Phi) \\ \sum_i w_i dy_i (\Phi_i - \Phi) \end{pmatrix} \quad (3.3.21)$$

$$\begin{pmatrix} \Phi_x \\ \Phi_y \end{pmatrix} = \begin{pmatrix} q_{11} & q_{12} \\ q_{21} & q_{22} \end{pmatrix} \begin{pmatrix} \sum_i w_i dx_i (\Phi_i - \Phi) \\ \sum_i w_i dy_i (\Phi_i - \Phi) \end{pmatrix} \quad (3.3.22)$$

$$\begin{pmatrix} \Phi_x \\ \Phi_y \end{pmatrix} = \begin{pmatrix} q_{11} \sum_i w_i dx_i (\Phi_i - \Phi) + q_{12} \sum_i w_i dy_i (\Phi_i - \Phi) \\ q_{21} \sum_i w_i dx_i (\Phi_i - \Phi) + q_{22} \sum_i w_i dy_i (\Phi_i - \Phi) \end{pmatrix}. \quad (3.3.23)$$

By comparing both sides row wise, we have

$$\Phi_x = q_{11} \sum_i w_i dx_i (\Phi_i - \Phi) + q_{12} \sum_i w_i dy_i (\Phi_i - \Phi), \quad (3.3.24)$$

$$\Phi_y = q_{21} \sum_i w_i dx_i (\Phi_i - \Phi) + q_{22} \sum_i w_i dy_i (\Phi_i - \Phi). \quad (3.3.25)$$

Now consider the equation (3.3.24)

$$\Phi_x = q_{11} \sum_i w_i dx_i (\Phi_i - \Phi) + q_{12} \sum_i w_i dy_i (\Phi_i - \Phi)$$

$$\Phi_x = \sum_i w_i (q_{11} dx_i + q_{12} dy_i) \Phi_i - \sum_i w_i (q_{11} dx_i + q_{12} dy_i) \Phi \quad (3.3.26)$$

Equation (3.3.26) can be written as

$$\Phi_x = a_1 + \Phi a_2 \quad (3.3.27)$$

with

$$a_1 = \sum_i w_i (q_{11} dx_i + q_{12} dy_i) \Phi_i$$

and

$$a_2 = - \sum_i w_i (q_{11} dx_i + q_{12} dy_i).$$

Similarly from the equation (3.3.25), we get

$$\Phi_y = b_1 + \Phi b_2 \quad (3.3.28)$$

with

$$b_1 = \sum_i w_i (q_{21} dx_i + q_{22} dy_i) \Phi_i$$

and

$$b_2 = - \sum_i w_i (q_{21} dx_i + q_{22} dy_i).$$

From the Eikonal equation (3.1.1), we have

$$\Phi_x^2 + \Phi_y^2 = F^2. \quad (3.3.29)$$

If we substitute (3.3.27) and (3.3.28) in (3.3.29), we get

$$(a_2^2 + b_2^2) \Phi^2 + 2(a_1 a_2 + b_1 b_2) \Phi + a_1^2 + b_1^2 = F^2 \quad (3.3.30)$$

which is a quadratic equation in  $\Phi$ , which yields two solutions for  $\Phi$ . As explained in Section 3.3.2, we choose the appropriate solution among them.

### 3.3.4 Algorithm

In this section, we present a detailed algorithm to solve the Eikonal equation. We follow the idea of front marching from fast marching method [73], but our method differs from fast marching method in approximating the solution. As in the fast marching method, divide all nodes in the domain into three categories, namely *Accepted*, *Narrow band* and *Far away*. Nodes, where the solution is known are named as *Accepted* nodes, all close neighbours of *Accepted* nodes in the range of radius  $h$  are named as *Narrow band* nodes and remaining nodes are named as *Far away* nodes.

Assume that, there are  $N$  number of nodes in the domain  $\Omega$  distributed randomly or uniformly. Given the initial front  $\Gamma$ , the algorithm to find the solution of the Eikonal equation in  $\Omega$  is explained below:

1. Initialization : Tag all nodes in the initial front as *Accepted*. Tag all neighbors of the initial front nodes as *Narrow band*. Tag all other nodes as *Far away*.
2. Assign  $\Phi = 0$  for all nodes in the initial front.
3. For each node in the *Narrow band*, compute  $\Phi$  using the method explained above.

4. Determine  $X_{min}$  such that  $\Phi$  has minimum value at  $X_{min}$  in *Narrow band*.
5. For each neighbour ' $X_i$ ' of  $X_{min}$ :
  - If  $X_i$  is in *Far away*, then compute  $\Phi(X_i)$  and add  $X_i$  to *Narrow band*
  - If  $X_i$  is already in *Narrow band*, then recompute  $\Phi(X_i)$
6. Remove  $X_{min}$  from *Narrow band* and add it to *Accepted*.
7. Repeat the steps 4-6 till all the nodes are *Accepted*.

### 3.3.5 Time Complexity

Let  $N$  be the number of nodes in the domain  $\Omega$  and  $n$  be the maximum number of *Accepted* neighbors for any node for given radius  $h$ . Call the process from step 4 to step 6 in the above algorithm as an iteration. During the algorithm, there will be maximum  $N$  iterations, since exactly one node is removed from the *Narrow band* for each iteration. For each iteration, during step 4, a search is performed for the smallest value of  $\Phi$  in the *Narrow band*. We use binary heap to store the *Narrow band* nodes and corresponding solution values as in fast marching method. So if there are  $M$  elements in *Narrow band*, then the process of finding minimum and ordering heap can be done in  $O(\log M)$  steps in worst case. Since  $M$  is bounded by  $N$ , the search is bounded by  $O(\log N)$ . For each iteration, step 5 will be performed for each neighbour, so step 5 will run at most  $n$  times. While approximating the solution, sorting of the *Accepted* neighbors with respect to the angles takes  $O(n \log n)$  in worst case, so the time complexity for step 5 will be  $O(n^2 \log n)$ . The total computational cost is bounded by  $N(\log(N) + n^2 \log n)$ .

Since  $n$  is smaller compared to  $N$ , computational cost to approximate the solution in the whole domain is  $O(N \log N)$ .

### 3.3.6 Numerical Examples

In this section, we demonstrate the performance of the our proposed method through few typical two dimensional examples, taken from the literature



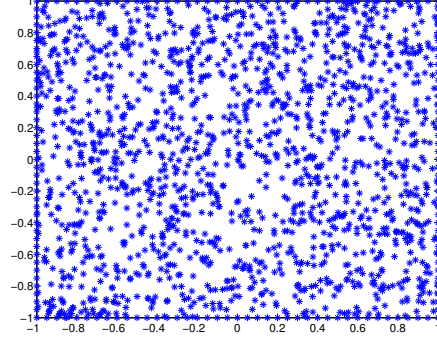


Figure 3.4: Randomly generated grid points

[55, 13]. To measure the accuracy, the following error functions are defined as in [26]

$$\text{Root Mean square error: } RMS = \frac{1}{N} \sqrt{\sum_{i=1}^N \left( \frac{\Phi_c(i) - \Phi_a(i)}{\Phi_a(i)} \right)^2} \quad (3.3.31)$$

$$\text{Average absolute error: } aerr = \sqrt{\frac{1}{N} \sum_{i=1}^N \left( \Phi_c(i) - \Phi_a(i) \right)^2} \quad (3.3.32)$$

$$\text{Relative error: } rerr = \sqrt{\frac{\sum_{i=1}^N (\Phi_c(i) - \Phi_a(i))^2}{\sum_{i=1}^N (\Phi_a(i))^2}} \quad (3.3.33)$$

where  $N$  is the number of nodes in the domain and  $\Phi_a$  is analytical solution and  $\Phi_c$  is the solution computed from the proposed method. In our numerical examples, we computed the solution for both cartesian and unstructured grids and compared with the analytical solution. We also presented error values for both cartesian and unstructured grids. A sample random grid is presented in Fig 3.4.

In case of the cartesian grid, the radius  $h$  is taken as  $2.5 \times dx$ , where  $dx$  is the step size in  $x$  and  $y$  directions. In case of the unstructured grid, by altering the radius ' $h$ ', we generate approximately same number of random points as in the cartesian grid.

**Example 1**

As the first example, we consider the Eikonal equation (3.1.1) with  $\Omega = [-1, 1] \times [-1, 1]$ ,  $\Gamma = \{(0, 0)\}$  and  $F(x, y) = 1$ . The exact solution

$$\Phi(x, y) = \sqrt{x^2 + y^2} \quad (3.3.34)$$

is the distance function from  $\Gamma$ .

In Table 3.1, the RMS error, the relative error and the average error for cartesian grid are presented. For unstructured grid, the error values are presented in Table 3.2. In Fig 3.5, the numerical solution for cartesian and unstructured grids and exact solution are presented. The Tables 3.1 and 3.2 show that accuracy improves as the number of nodes increases.

Table 3.1: Error values for Example 1: Cartesian Grid

dx	N	RMS	rerr	aerr
0.1	441	7.93E-05	1.30E-03	1.17E-03
0.05	1681	2.65E-05	8.51E-04	7.16E-04
0.025	6561	8.57E-06	5.32E-04	4.42E-04
0.0125	25921	2.67E-06	3.23E-04	2.61E-04

Table 3.2: Error values for Example 1: Unstructured Grid

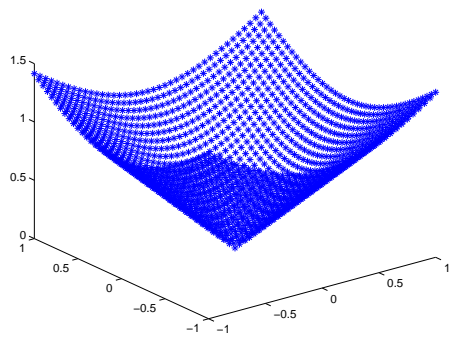
h	N	RMS	rerr	aerr
0.28	492	8.56E-05	1.43E-03	1.30E-03
0.14	1743	3.28E-05	1.05E-03	5.73E-04
0.07	6413	1.24E-05	6.73E-04	3.02E-04
0.035	24786	4.05E-06	4.34E-04	1.91E-04

**Example 2:**

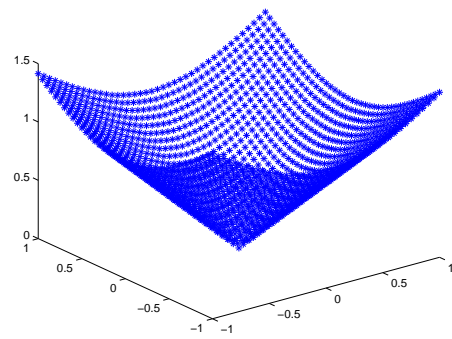
For the second example, consider the Eikonal equation (3.1.1) with  $\Omega = [-1, 1] \times [-1, 1]$ ,  $F(x, y) = 1$ .  $\Gamma$  is circle with center(0,0) and the radius  $R = 0.5$ . The exact solution of this problem is the distance function from  $\Gamma$ ,

$$\Phi(x, y) = \|\sqrt{x^2 + y^2} - R\| \quad (3.3.35)$$

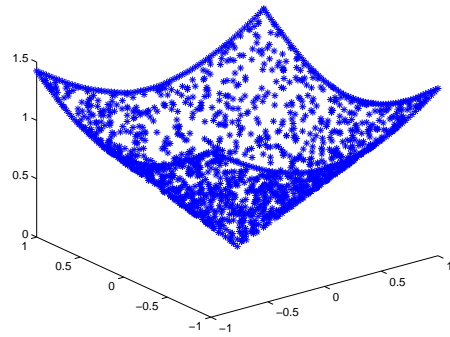
The error values are presented in Table 3.3 and Table 3.4 for cartesian and unstructured grids respectively. The graph of the numerical and the analytical solutions are plotted in Fig 3.6.



(a) Analytical



(b) Numerical-cartesian



(c) Numerical-random

Figure 3.5: Solution description of Example 1

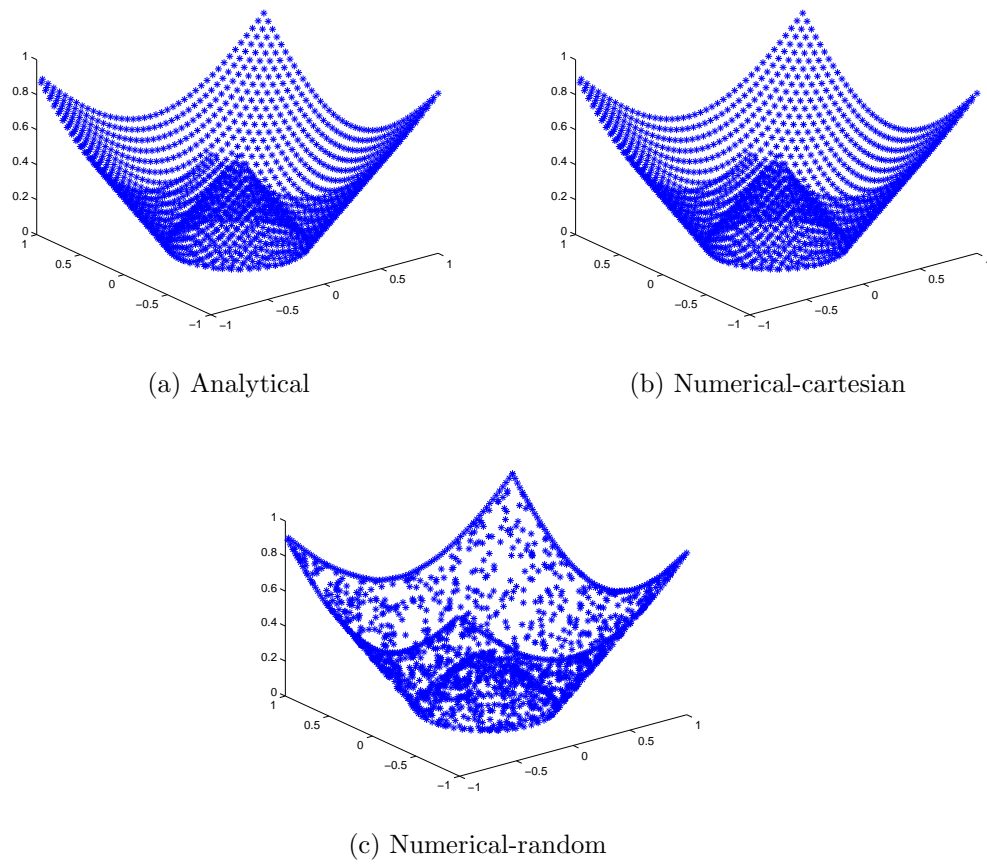


Figure 3.6: Solution description of Example 2

Table 3.3: Error values for Example 2: Cartesian Grid

dx	N	RMS	rerr	aerr
0.1	441	9.34E-03	1.45E-02	5.95E-03
0.05	1681	3.82E-03	1.05E-02	4.24E-03
0.025	6561	1.29E-03	6.37E-03	2.57E-03
0.0125	25921	9.03E-04	2.02E-03	8.02E-04

Table 3.4: Error values for Example 2: Unstructured Grid

h	N	RMS	rerr	aerr
0.28	492	9.82E-03	1.83E-02	8.19E-03
0.14	1743	5.66E-03	1.14E-02	2.31E-03
0.07	6413	1.39E-03	6.46E-03	2.07E-03
0.035	24786	9.53E-04	1.66E-03	6.72E-04

**Example 3**

Consider the Eikonal equation (3.1.1) when  $\Omega = [-1, 1] \times [-1, 1]$ ,  $\Gamma = \partial\Omega$  and

$$F(x, y) = 2\sqrt{x^2(1 - y^2) + y^2(1 - x^2)}. \quad (3.3.36)$$

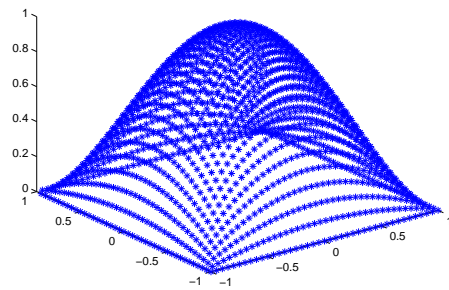
The exact solution is

$$\Phi(x, y) = (1 - x^2)(1 - y^2). \quad (3.3.37)$$

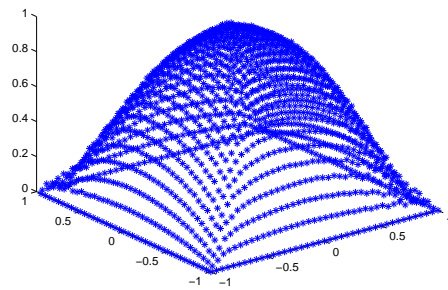
The RMS, relative and average errors are presented in Table 3.5 and Table 3.6 for cartesian and unstructured grids respectively. The plot of the numerical solutions and the analytical solution are depicted in Fig 3.7.

Table 3.5: Error values for Example 3: Cartesian Grid

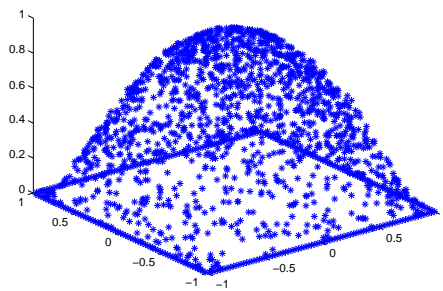
dx	N	RMS	rerr	aerr
0.1	441	1.61E-03	9.04E-03	5.87E-03
0.05	1681	8.60E-04	6.46E-03	5.45E-03
0.025	6561	4.54E-04	1.07E-03	5.03E-03
0.0125	25921	2.32E-04	9.06E-04	4.24E-03



(a) Analytical



(b) Numerical-cartesian



(c) Numerical-random

Figure 3.7: Solution description of Example 3

Table 3.6: Error values for Example 3: Unstructured Grid

h	N	RMS	rerr	aerr
0.28	492	3.39E-03	2.64E-02	1.46E-02
0.14	1743	1.29E-03	1.15E-02	6.05E-03
0.07	6413	6.18E-04	8.61E-03	4.56E-03
0.035	24786	2.87E-04	7.49E-03	4.42E-03

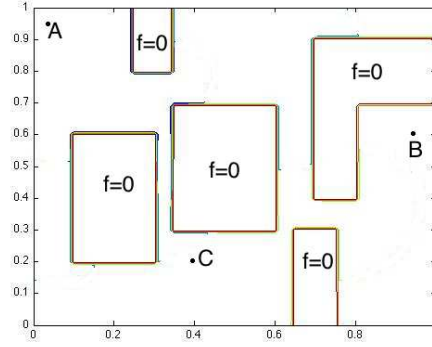


Figure 3.8: Computational Domain

#### Example 4: Path Planning

In this example, an application of the solution of the Eikonal equation in path planning is presented. We consider a two-dimensional path planning problem with constraints in a domain  $\Omega = [0, 1] \times [0, 1]$  with obstacles as shown in Fig 3.8. Given starting points  $B(0.95, 0.6)$  and  $C(0.4, 0.2)$  and end point  $A(0.1, 0.9)$ , our goal is to find the shortest path.

The function  $F(x, y) = 1/f(x, y)$ , where

$$f(x, y) = \begin{cases} 0, & \text{if } (x, y) \in Z \\ 1 & \text{otherwise} \end{cases}$$

with  $Z = ([0.7, 1.0] \times [0.7, 0.9]) \cup ([0.65, 0.75] \times [0, 0.3]) \cup ([0.35, 0.6] \times [0.3, 0.7]) \cup ([0.25, 0.35] \times [0.8, 1.0]) \cup ([0.1, 0.3] \times [0.2, 0.6]) \cup ([0.7, 0.8] \times [0.4, 0.7])$ . We first approximate the solution of the Eikonal equation with  $\Gamma = \{A\}$ , then we backtrack the shortest path from the starting points B and C along with the direction of negative gradient of  $\Phi$ . The contour plot of the solution of Eikonal equation and shortest paths from points B and C to point A are presented in Fig 3.9.

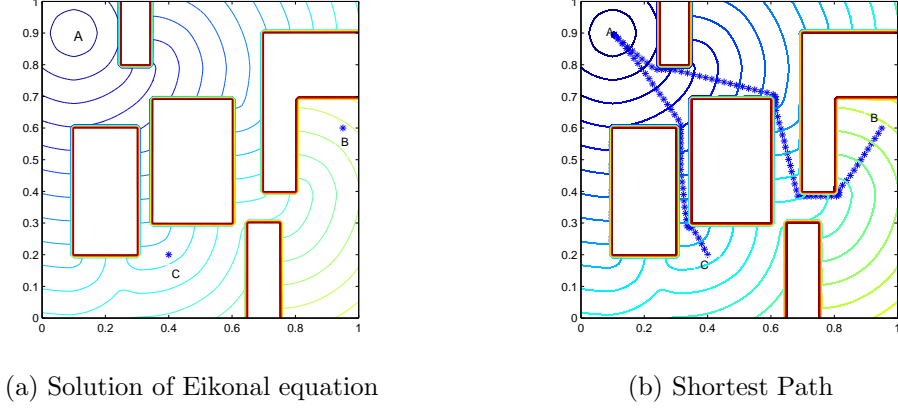


Figure 3.9: Path Planning

In all the test problems presented here, the values of RMS, relative and average errors showed that the proposed method is accurate with different number of node points for both cartesian and unstructured grids.



# Chapter 4

## Numerical Method

This chapter demonstrates the simulation procedure of micro, hydro as well as scalar models and the numerical scheme we used. The microscopic flow simulation is explained in Section 4.1 and in section 4.2 the simulation of hydro and scalar models are explained. The particle method we used to solve the governing equations is presented briefly in section 4.3 .

### 4.1 Microscopic Simulation

The microscopic equations (2.3.1) are solved by explicit Euler method for system of ordinary differential equations. For a straightforward implementation, the complete distance matrix  $(d_{ij}) = |x_i - x_j|$  has to be computed in order to evaluate the interaction potential of the microscopic equations. If there are many particles in the system, then computing the distance matrix is computationally expensive. To overcome this, one can use nearest neighbor lists for short range interactions or Greengard-Rohklin type algorithms for long range interactions. Since pedestrians interact with only pedestrians who are nearby we use an implementation with a nearest neighbor list reducing the computational cost considerably.

We divide the computational domain into cells. Each cell is of same area and is represented by its centre. The macroscopic quantities  $\rho, u$  are evaluated at these cell centres. Density  $\rho$  is the number of pedestrians in the cell divided by  $N_{max}$ , where  $N_{max}$  is the maximum number of pedestrians in one cell and  $u$  is average velocity of pedestrians in the cell. Pedestrians in each cell will interact with only pedestrians in the neighboring cells. The

grid containing all cell centres is used to compute the solution of the Eikonal equation.

## 4.2 Hydrodynamic and Scalar Simulation

The hydrodynamic limit equations are considered in detail numerically using a macroscopic particle method see Ref. [81]. The particle method is based on a Lagrangian formulation of the hydrodynamic equations (2.4.1, 2.4.3). We consider

$$\begin{aligned}\frac{dx}{dt} &= u \\ \frac{d\rho}{dt} &= -\rho \frac{\partial u}{\partial x} \\ \frac{du}{dt} &= \hat{G}(\rho, \Phi, u) + \hat{F}(\rho, u),\end{aligned}$$

One evaluates these quantities at the particle locations and approximates the spatial derivatives of  $u$  by a difference approximation, explained in the section (4.3). The integral over the interaction potential is evaluated by a straightforward integration rule:

$$\hat{F}(\rho, u) \sim \sum_j F(x - x_j, u(x) - u_j) \rho_j dV_j,$$

where  $dV_j$  is the local area around a particle determined by a nearest neighbour search. The resulting equations are then solved by a time discretization of arbitrary order. Diffusive terms can be included as well in a straightforward way. Obviously, this shows that the actual macroscopic computations are very similar to the microscopic ones. The difference lies in the way the interaction term is evaluated. In the microscopic case we compute

$$\frac{1}{N} \sum_j F(x - x_j, v - v_j)$$

instead of the above expression. If the values of  $\rho_j$  and  $dV_j$  are all equal then using

$$1 = \int \rho(x) dx \sim \sum_j \rho_j dV_j$$

it is easy to see that both simulations are equivalent to each other. However, in the macroscopic situation the particles are not physical particles as in the

microscopic case. They play the role of discretization points. In particular, if the number of 'real' particles is very large, that does not mean that the number of macroscopic particles in the particle method has to be increased in the same way. The number of macroscopic particles is only chosen according to accuracy considerations. On the other hand, the macroscopic equations considered here are derived under the assumption of a mono-kinetic distribution function. Thus, they are not able to capture all microscopic patterns.

Finally, the diffusive equations are solved as well with a particle method. In this case the so called diffusion velocity methods is used, i.e., the convection diffusion equation are written as a pure transport problem

$$\partial_t \rho + \nabla_x \cdot (a \rho) = 0 \quad (4.2.1)$$

with

$$a = -f(\rho(x)) \frac{\nabla \Phi(x)}{\|\nabla \Phi(x)\|} + TF \star \rho$$

and then solved in a Lagrangian way as usual.

In all cases the solution of the Eikonal equation is coupled to the flow simulation. The Eikonal equation is solved on separate structured or unstructured grid. The data obtained from this simulation are then interpolated onto particle locations used for flow simulation. On the other hand, to solve the Eikonal equation the density data from the flow simulation are needed and have to be interpolated onto the fixed grid for the Eikonal equation. We update the Eikonal equation in every tenth time step in order to save computational time.

### 4.3 Finite Pointset Method (FPM)

The basis of the computations in FPM is point cloud, which represents the flow field. The points of the cloud are referred to as particles or numerical grids. They are carriers of all relevant physical information. The particles have to completely cover the whole domain, i.e. the point cloud has to fulfil certain quality criteria (particles are not allowed to form holes which means particles have to find sufficiently many neighbors; also, particles are not allowed to cluster; etc.). The point cloud is a geometrical basis, which allows for a numerical formulation making FPM a general finite difference

idea applied to continuum mechanics. That especially means, if the point cloud would receive to a classical finite difference method. The idea of general finite difference also means that FPM is not based on a weak formulation like the Galerkin's approach. Rather, FPM is a strong formulation which models differential equations by direct approximation of the occurring differential operators. The method used is a moving least square idea which was specially developed for FPM.

### 4.3.1 Least Square Approximation

In this section, we present the least square approximation of a function at an arbitrary position from the surrounding cluster of points.

Consider the computational domain  $\Omega \subset R^n, n = 1, 2, 3$ . Consider a set of clouds, grids or particles  $P(x, h) = \{x_i : x_i \in \Omega, i = 1, \dots, N\}$ . The distribution of particles does not necessarily have to be uniform and can be quite arbitrary. A typical distribution of particles, for example in 2D, looks like in Fig. 4.1.

Let  $f(x)$  be a scalar function and  $f_i$  its values at  $x_i$  for  $i = 1, 2, \dots, N$ .

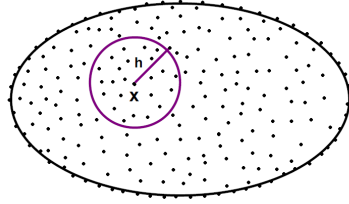


Figure 4.1: Flow domain with non-structured grid

Consider the problem to approximate  $f(x)$  and its spatial derivatives at  $x$  in terms of the values of a set of neighbouring points. In order to limit the number of points we associate a weight function  $w = w(x_i - x; h)$  with a small compact support, where  $h$  determines the size of the support, as shown in Fig. 4.1. In FPM,  $h$  is known as a smoothing length. The smoothing length defines a set of neighbouring particles around  $x$ . The weight function can be quite arbitrary but in our computations, we consider a Gaussian weight function in the following form

$$w(x_i - x; h) = \begin{cases} \exp(-C \frac{\|x_i - x\|^2}{h^2}), & \text{if } \frac{\|x_i - x\|}{h} \leq 1 \\ 0, & \text{otherwise} \end{cases}$$

where  $C$  a positive constant is considered to be in the range of 2 to 6 and depends upon the users.

For consistency reasons some obvious restrictions are required, namely for example, in  $2D$ , if we want the second order approximation there should be at least 6 neighbour particles including the central particle and they should neither be on the same line nor on the same circle. Hence we define the size of  $h$  such that the minimum number of neighbours is guaranteed for the approximation of derivatives. Hence new particles will have to be introduced into the simulations as the particle distribution becomes too sparse or particles will have to be removed from the computation as they become too dense.

Let  $N(x, h) = \{x_i : i = 1, 2, \dots, m\}$  be the set of  $m$  neighbouring points of  $x$ . We approximate the function  $f(x)$  by  $f_h(x)$  as  $f_h(x) = \sum_{i=1}^N f_i \phi(x_i, x)$ , where the shape function  $\phi(x_i, x)$  is computed at each point  $x$  by the least square method over its own compact support. We note that  $\phi_i = 0$  for all  $i \notin N(x, h)$ . It is important to stress that this expression is consistent only if the function  $\phi$  is 1 at  $x_i$ , namely  $\phi(x_i, x_j) = \delta_{ij}$  for all  $i, j = 1, 2, \dots, N$ .

The approximation of the first and second order derivatives can be computed directly from  $f_h(x)$  or directly by using the least square method. The first method is known in literature as moving least square method [24], [54]. Usually the function  $f_h(x)$  and its derivatives  $f_{kh}(x)$  are not smooth enough to be differentiable and therefore the second order derivatives can not be properly computed. We approximate the derivatives  $\partial f(x)/\partial x_k$  by  $f_{kh}(x) = \sum_{i=1}^N f_i \eta_k(x_i, x)$  for  $k = 1, 2, 3$ , where  $\eta_k(x_i, x)$  is directly computed by the least squares interpolation. In a similar manner we define the approximation for the second order derivatives  $\partial^2 f(x)/\partial x_l \partial x_k$  by  $f_{klh}(x) = \sum_{i=1}^N f_i \Psi_{kl}(x_i, x)$  for  $k, l = 1, 2, 3$ . The determination of the function  $f(x)$ ,  $f_{kh}(x)$  and  $f_{klh}(x)$  for  $k, l = 1, 2, 3$  can be computed easily and accurately by using the Taylor series expansion and the least square approximation. We write a Taylor's expansion around the point  $x$  with unknown coefficients by minimizing a weighted error over the neighbouring points. The optimization is constrained to satisfy  $\phi(x_1, x_1) = 1$  where  $x_1$  is the closest point, namely the approximation must interpolate the closest point.

In order to approximate the function and its derivatives at  $x$  by using a quadratic approximation through the  $m$  neighboring points sorted with re-

spect to its distance from  $x$  we let

$$\begin{aligned} f(x_i) &= f_h(x) + \sum_{k=1}^3 f_{kh}(x)(x_{ki} - x_k) \\ &+ \frac{1}{2} \sum_{k,l=1}^3 f_{klh}(x)(x_{ki} - x_k)(x_{li} - x_k) + e_i, \end{aligned} \quad (4.3.1)$$

where  $e_i$  is the error in the Taylor's expansion at the point  $x_i$ . The unknowns  $f_h, f_{kh}$  and  $f_{klh}$  for  $k, l = 1, 2, 3$  are computed by minimizing the error  $e_i$  for  $i = 2, 3, \dots, m$  and setting the constraint  $e_1 = 0$ . By subtracting the first equation with  $e_1 = 0$  to all the other equations the system can be written as  $e = Ma - b$ , where  $M$  is the  $m \times 9$  geometrical matrix, in  $3D$ , whose components consist of the directional distances from the  $m$ -neighbours to the central particle  $x$ . The other vectors are denoted by  $a = [f_{1h}, f_{2h}, f_{3h}, f_{11h}, f_{12h}, f_{13h}, f_{22h}, f_{23h}, f_{33h}]^T, b = [f_2 - f_1, f_3 - f_1, \dots, f_m - f_1]^T, e = [e_2, e_3, \dots, e_m]^T$ . For  $m > 9$ , this system is over-determined for the nine unknowns  $f_{kh}$  and  $f_{klh}$  for  $k, l = 1, 2, 3$ .

The unknowns  $a$  are obtained from a weighted least squares method by minimizing the quadratic form

$$J = \sum_{i=1}^m w_i e_i^2. \quad (4.3.2)$$

The above equations can be expressed in the form

$$J = (Ma - b)^T W (Ma - b), \quad (4.3.3)$$

where  $W = \delta_{ij} w_i$ . The minimization of  $J$  formally yields

$$a = (M^T W M)^{-1} (M^T W) b. \quad (4.3.4)$$

Now from the equation for the closest point  $x_1$  we can compute the value of  $\Psi(x)$  at  $x$  as

$$f_h(x) = f(x_1) - \sum_{k=1}^3 f_{kh}(x)(x_{k1} - x_k) - \frac{1}{2} \sum_{k,l=1}^3 f_{klh}(x)(x_{k1} - x_k)(x_{l1} - x_k) \quad (4.3.5)$$

since  $f_{kh}$  and  $f_{klh}$  for  $k, l = 1, 2, 3$  are now known. The solution of the constrained least squares problem is straightforward and more sophisticated

techniques can be used. For example, the minimization or singular decomposition techniques can be very helpful to determine efficiently the unknowns.

We note that if the approximation is computed at  $x_i$  we have  $f_h(x_i) = f_i$  which implies  $\phi(x_i, x_j) = \delta_{ij}$  can be approximated very closely performing the unconstrained least squares minimization over all the  $m$  equations.





# Chapter 5

## Numerical Results

In this chapter, we present two numerical examples. In the first numerical example, we present a series of numerical experiments on microscopic, hydrodynamic and scalar models. We compare our models for different parameters and for different types of obstacles. In the second numerical example, we are interested in comparing our results with the experimental results.

### 5.1 Numerical Example 1

In this section we present few numerical experiments on the microscopic equations (2.3.1), as well as the hydrodynamic (2.4.1, 2.4.3) and scalar limit (2.5.3). Different situations are studied and various patterns are investigated.

#### 5.1.1 Initialization

We investigate the models numerically for a configuration defined in Ref. [56]. Consider a railway platform of  $100m$  length and  $50m$  width with an obstruction in the middle. We consider two types of obstructions in our simulations, circular and square obstacles as shown in Figures 5.1 and 5.2, respectively.

Pedestrians enter the platform from the left boundary and leave at the either of the two exits of  $15m$  width on the right boundary. Initially the platform is empty. The density of the pedestrians increases from linearly from zero to a peak at  $t = 60sec$ , and then drops linearly back to zero at

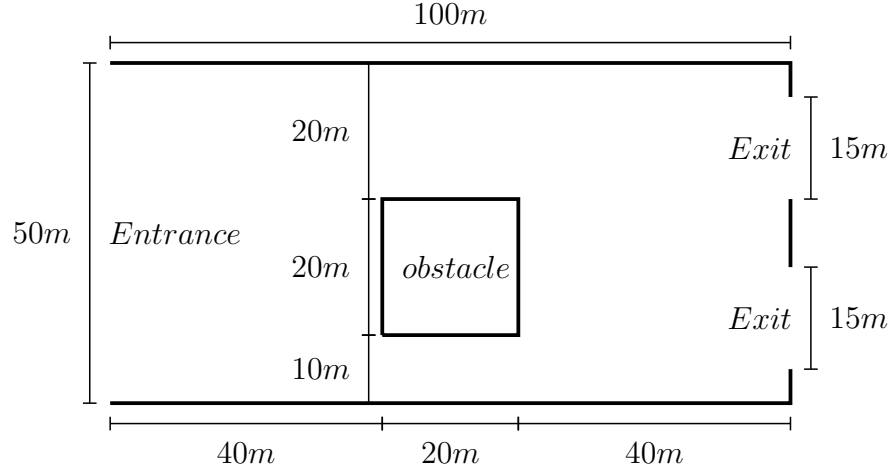


Figure 5.1: The railway platform with a square obstruction.

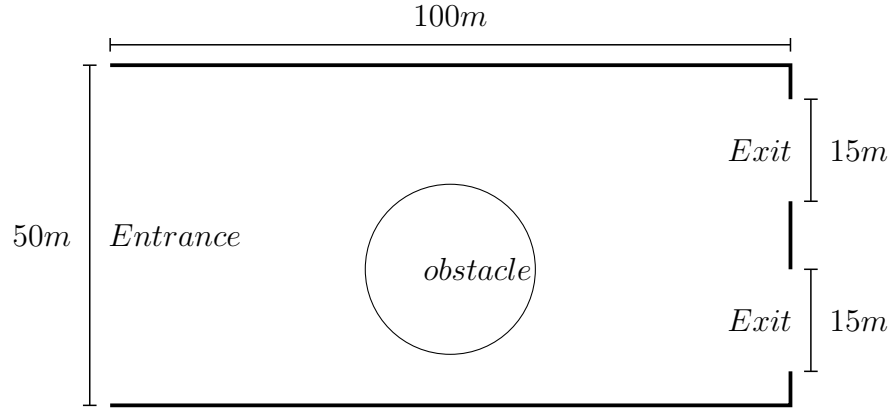


Figure 5.2: The railway platform with a circular obstruction.

$t = 120\text{sec}$ , and no pedestrians enter the platform thereafter. It is given by

$$\rho = \begin{cases} 2.5 \frac{t}{60}, & \text{if } 0 \leq t \leq 60 \\ -\frac{2.5}{60}(t - 120), & \text{if } 60 \leq t \leq 120 \\ 0, & \text{else} \end{cases} \quad (5.1.1)$$

We use two types of grid points, one for solving the pedestrian flow model and another for solving the Eikonal equation. Therefore, we establish two clusters of grid points, which are decoupled from each other, however, we interchange the necessary information from one cluster grid to another and vice-versa. For solving the Eikonal equation, we use a fixed grid in case of

both square and circle obstructions.

In microscopic model, we divide the computational domain into  $N$  cells and each cell is represented by its centre. We define  $N_{max}$  as the maximum number of pedestrians in each cell. The density in the microscopic model at each cell is calculated by dividing the number of pedestrians in that cell by  $N_{max}$ , i.e.,

$$\rho = \frac{N}{N_{max}} \quad (5.1.2)$$

where  $N$  is the number of pedestrians in the cell. In our simulations, we generate  $\rho N_{max}$  pedestrians in the cells containing the left boundary, where  $\rho$  is defined as in (5.1.1). By altering  $N_{max}$ , we control the number of pedestrians in the simulation.

In hydrodynamic and scalar models, we initiate pedestrians on the left boundary with a distance  $2.1 \times R$ , such that they do not interact with each other in the initial stages. Here  $R$  is the interaction radius.

### 5.1.2 Boundary Conditions for the Eikonal equation

The computational domain  $[0, 100]m \times [0, 50]m$  is covered with a fixed grid to solve the Eikonal equation. The grid points inside the obstruction are excluded from the computation. The boundary conditions for the Eikonal equation are as follows.

- On the inflow boundary, the value of  $\Phi$  is not specified.
- On the solid wall boundary namely, the top, bottom and part of the right boundary and the boundary of the obstruction, the values of  $\Phi$  set as infinity.
- On the outflow boundary, namely at  $x = 100$ ,  $5 \leq y \leq 20$  and  $30 \leq y \leq 45$ , we set  $\Phi$  as zero.

### 5.1.3 Parameters

We choose the inflow velocity as  $(U(\rho), 0)$ , where  $U(\rho)$  is the speed-density relation is given by a linear dependence

$$U(\rho) = u_{max}(1 - \rho/\rho_{max}), \quad (5.1.3)$$

where  $u_{max}$  is the free flow velocity and  $\rho_{max}$  is the maximum density of pedestrians in dense areas. The other parameter values are taken as in Table 5.1.

Name of the paramter	Notation	Value
Relaxation parameter	T	0.001
Interaction radius	R	0.35
Interaction constant	$k_n$	$3e5$
Friction co-efficient	$\gamma_n$	10
Tangential co-efficient	$\gamma_t$	2
Free flow velocity	$u_{max}$	$2m/s$
Maximum density	$\rho_{max}$	$10ped/m^2$

Table 5.1: Parameters

We use an explicit time integration for solving the hydrodynamic and scalar models with the constant time step  $\Delta t = 0.00125$  for all cases.

### 5.1.4 Results

#### Computational Time

The computation times for a simulation of the microscopic model up to  $t = 150s$  are given for different numbers of particles in Table 5.2. We chose

$N_{max}$	RUN TIME	# Pedestrians
1	05:19	22300
2	08:25	37000
3	16:46	53400
4	23:48	67600
5	38:29	83100

Table 5.2: Microscopic model: Particles per cell  $N_{max}$ , computing times for  $t = 150s$  and maximal total number of pedestrians.

the cell size in microscopic model approximately same as the size of the macroscopic particle. i.e., the microscopic density is evaluated with a cell size given by  $dx = 0.7143$ . This "cell size" is also used on the average

Model	$dx = 1.0$	$dx = 0.7143$
Hydro	04:15	06:23
Scalar	04:44	07:36

Table 5.3: Hydrodynamic and scalar model: Computing times for  $t = 150s$  for  $dx = 1.0$  and  $dx = 0.7143$ .

for the macroscopic computation, which gives approximately  $N = 22000$  macroscopic particles.

The computational time for a simulation of hydrodynamic and scalar model up to  $t = 150s$  are presented in Table 5.3 for different  $dx$ . The macroscopic models give a good approximation of the microscopic problem, if at least 3 microscopic particles per cell are used, see Figure 5.3. This corresponds to a total number  $N = 53400$  of particles for the microscopic simulation. The results in Tables 5.2 and 5.3 show that the computation times for the macroscopic simulation are considerably smaller than for the microscopic one if 3 or more microscopic particles per cell are used.

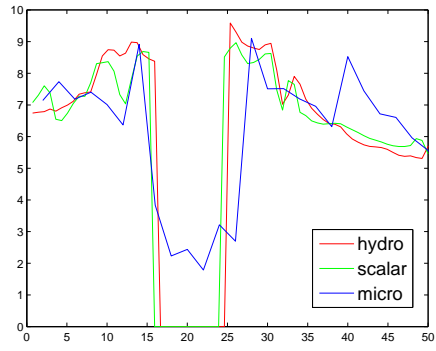
### Comparison between Micro, Hydro and Scalar models

For  $t = 120s$  the plots of the density are shown for microscopic, hydrodynamic and scalar model in Figure 5.4.

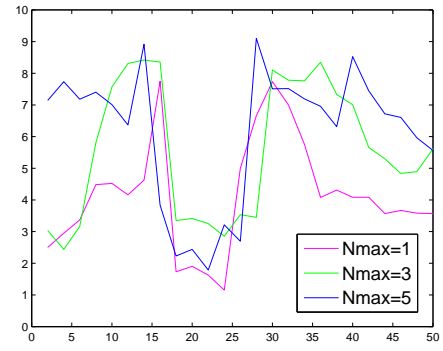
Figures 5.5, 5.6, 5.7 and 5.8 show the time evolution of the microscopic, the macroscopic and the scalar particles for time  $t = 25s$ ,  $t = 50s$ ,  $t = 100s$  and  $t = 150s$ , respectively. All three solutions show a very similar behaviour for the set of parameters chosen here.

Figure 5.9 shows the density in the hydrodynamic model and solution of the Eikonal equation for time  $t = 25s$ ,  $t = 50s$ ,  $t = 100s$  and  $t = 150s$ . Figure 5.9 illustrates the evolution of the congestion pattern on the railway platform. From the density plots in Figure 5.9, it is observed that two shocks are seen before the obstruction as density of pedestrians increase at time  $t = 100s$ . These kind of shocks are frequently observed in a walking facility with large crowds when pedestrians queue up to walk through a bottleneck with reduced capacity. Three triangular vacuum regions on two sides of obstacle and between the two exits are also observed which is because of travel cost minimization strategy of the pedestrians.

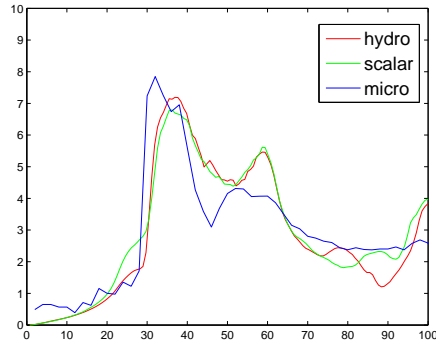
For a different set of parameters one can observe larger deviations between the hydrodynamic and scalar models. Choose, for example,  $T = 0.001$ ,



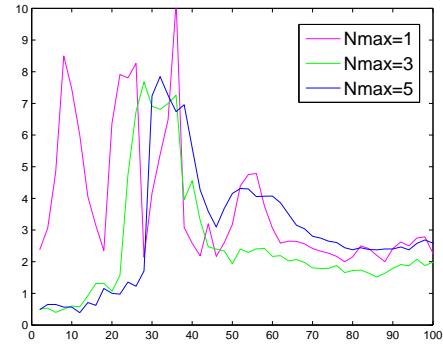
(a) cutx36-comparison



(b) cutx36-micro

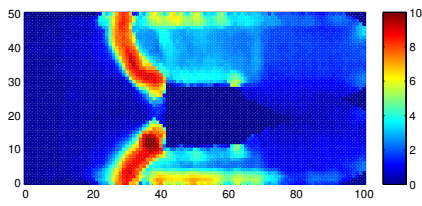


(c) cuty33-comparison

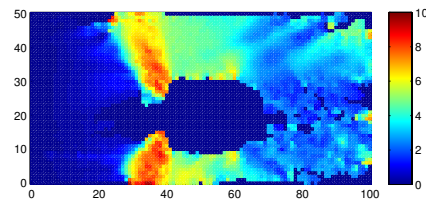


(d) cuty33-micro

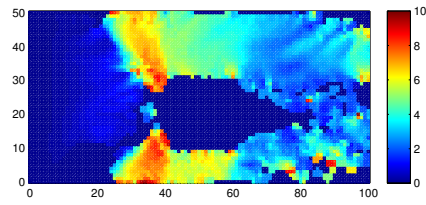
Figure 5.3: Density at  $t = 120$  along  $x = 36$  and  $y = 33$  for hydro-dynamic, scalar and microscopic models.



(a) Microscopic

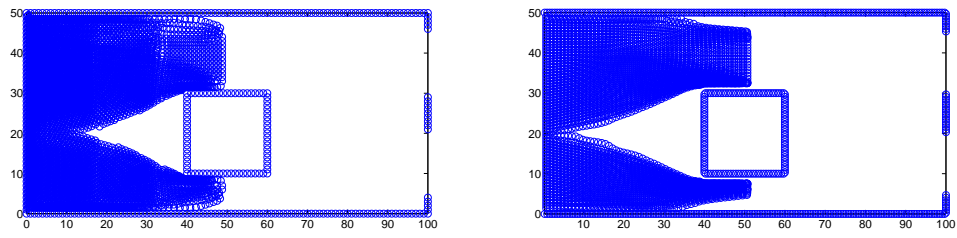


(b) Hydrodynamic



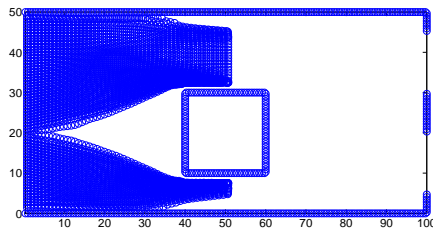
(c) Scalar

Figure 5.4: Density of hydrodynamic and scalar models at  $t = 120s$ .



(a) Microscopic

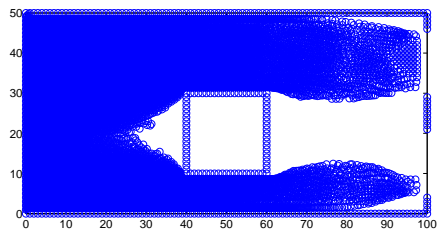
(b) Hydrodynamic



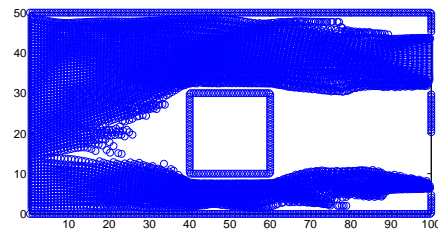
(c) Scalar

Figure 5.5: Distribution of particles for microscopic, hydrodynamic and simplified models at  $t = 25s$ .

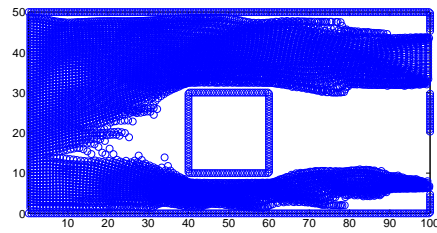




(a) Microscopic

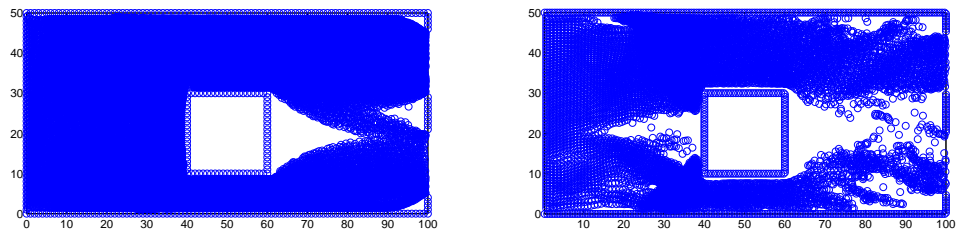


(b) Hydrodynamic



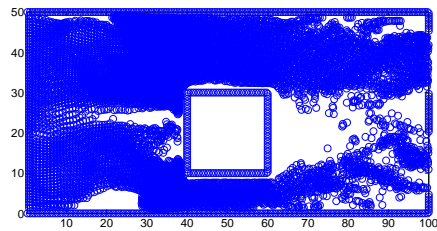
(c) Scalar

Figure 5.6: Distribution of particles for microscopic, hydrodynamic and simplified models at  $t = 50s$ .



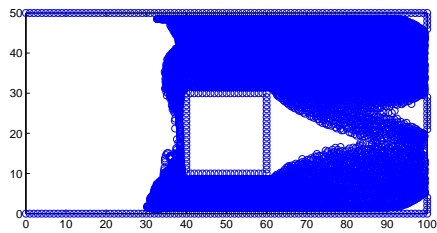
(a) Microscopic

(b) Hydrodynamic

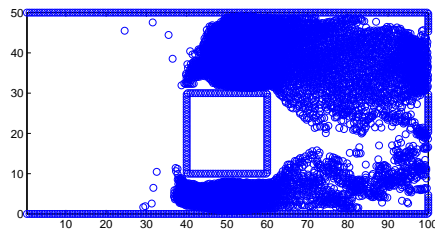


(c) Scalar

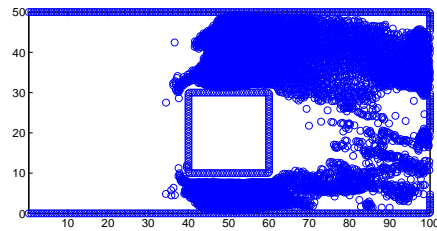
Figure 5.7: Distribution of particles for microscopic, hydrodynamic and simplified models at  $t = 100s$ .



(a) Microscopic



(b) Hydrodynamic



(c) Scalar

Figure 5.8: Distribution of particles for microscopic, hydrodynamic and simplified models at  $t = 150s$ .

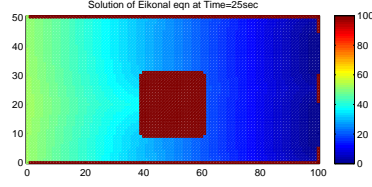
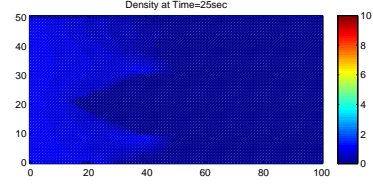
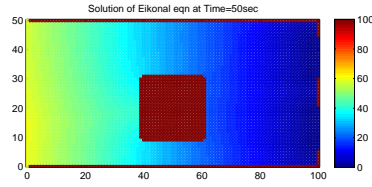
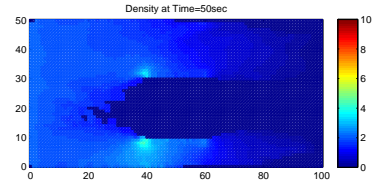
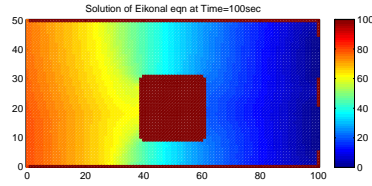
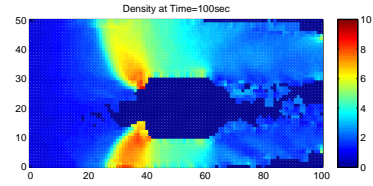
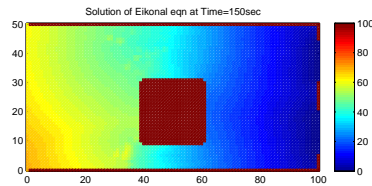
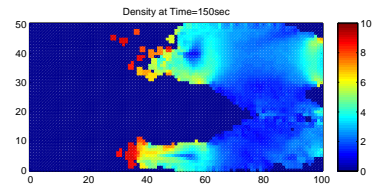
(a)  $t=25s$ (b)  $t=25s$ (c)  $t=50s$ (d)  $t=50s$ (e)  $t=100s$ (f)  $t=100s$ (g)  $t=150s$ (h)  $t=150s$ 

Figure 5.9: Solution of the Eikonal equation and density for  $t = 25s$ ,  $t = 50s$ ,  $t = 100s$  and  $t = 150s$ .

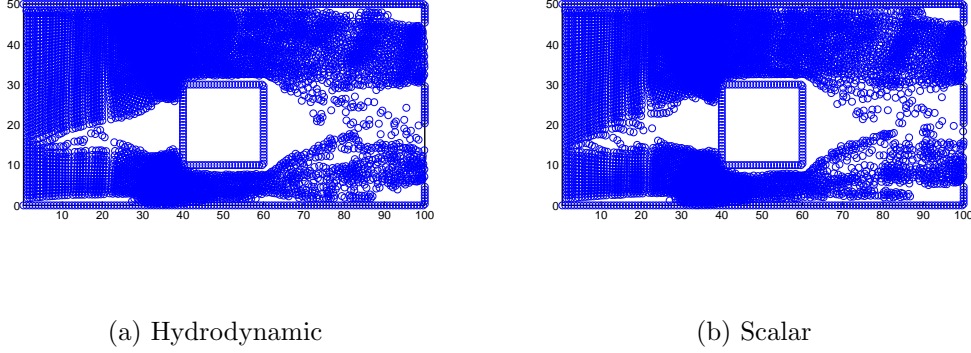


Figure 5.10: Distribution of particles for hydrodynamic and scalar model at  $t = 100s$  for relaxation time  $T = 0.001$ .

$T = 0.01, T = 0.1$  and  $T = 1.0$  and the other parameters as  $k_n = 1000$ ,  $\gamma_n = 1.0$  and  $\gamma_t = 0.2$ . This leads to the distribution of particles at  $t = 100s$  shown in Figures 5.10, 5.11, 5.12 and 5.13.

### Comparison with square and circle obstacles

Now, we investigate obstacles with smooth boundaries, e.g. a circular obstacle and compare it with square obstacle. We consider a circular and rectangular obstacles of equal area. We present the distribution of pedestrians at  $t = 150s$  and  $t = 175s$  obtained from the hydrodynamic model in Figures 5.14 and 5.15 respectively. The solution of the Eikonal equation for the two obstructions for  $t = 25s$ ,  $t = 50s$ ,  $t = 100s$  and  $t = 150s$  are presented in Figure 5.16 and comparison of density for two cases is presented in Figure 5.17 for  $t = 25s$ ,  $t = 50s$ ,  $t = 100s$  and  $t = 150s$ . From all these results, we observe that, even if the area of the obstacles is equal, a smooth obstacle leads to smaller evacuation times than a rectangular obstacle. Smooth obstacle yield smoother and lower density distribution and provides a better walking environment.

### Multiple obstacles

Finally, to study the pedestrian flow in complex geometry, we consider a railway platform with multiple obstacles. Consider the railway platform with multiple obstacles as shown in Figure 5.18. Pedestrians enter from the en-

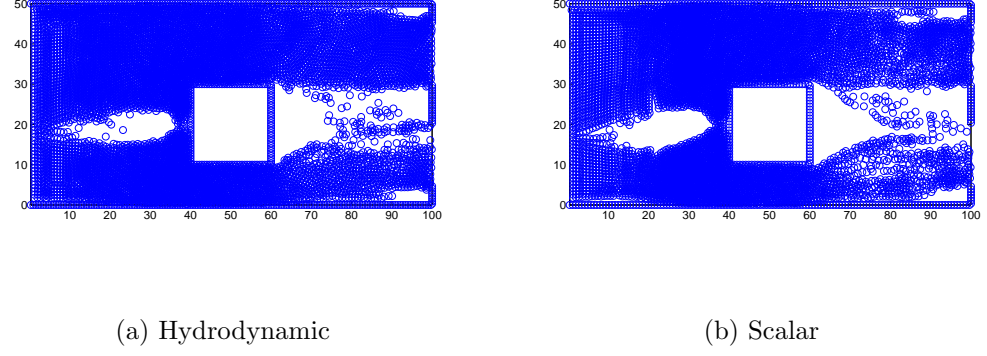


Figure 5.11: Distribution of particles for hydrodynamic and scalar model at  $t = 100s$  for relaxation time  $T = 0.01$ .

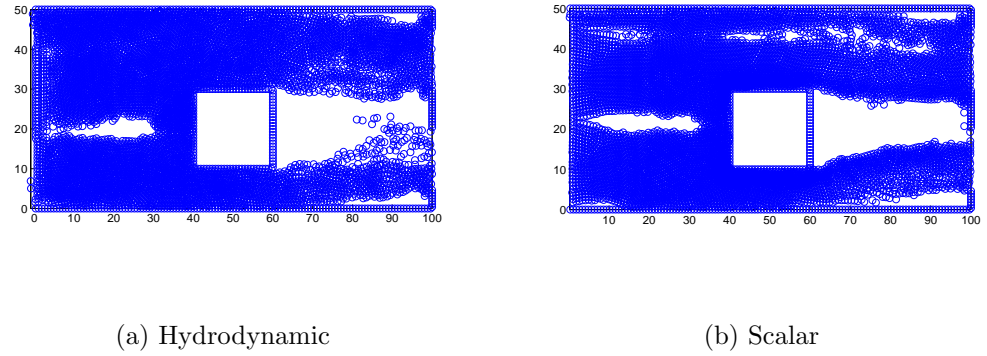


Figure 5.12: Distribution of particles for hydrodynamic and scalar model at  $t = 100s$  for relaxation time  $T = 0.1$ .

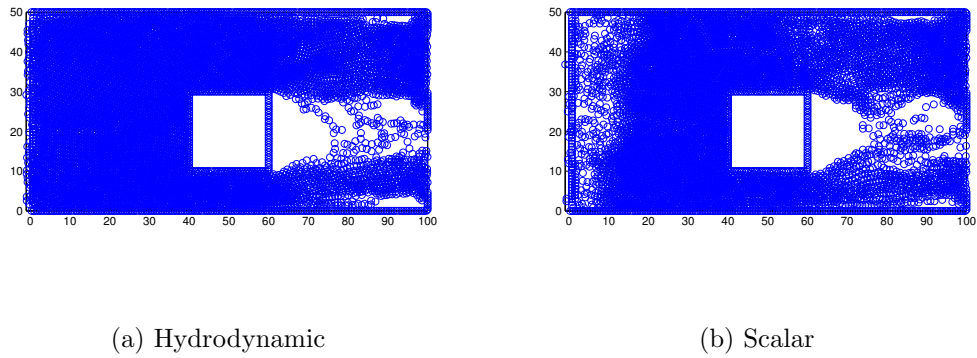


Figure 5.13: Distribution of particles for hydrodynamic and scalar model at  $t = 100s$  for relaxation time  $T = 1.0$ .

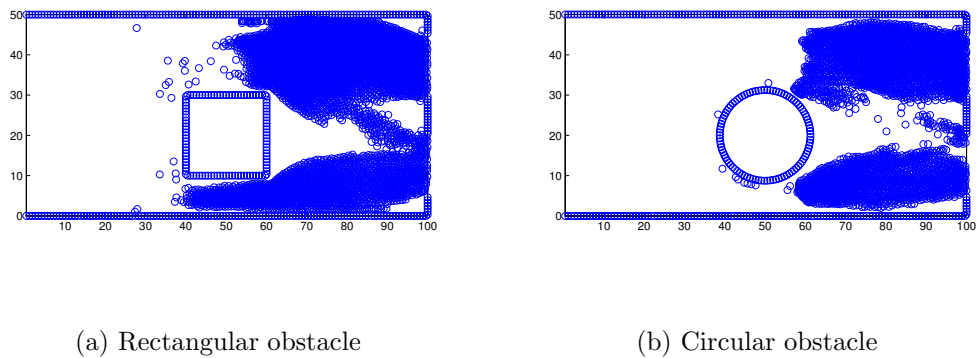


Figure 5.14: Comparison of circular and rectangular obstacles for  $t = 150s$  for relaxation time  $T = 0.001$ .

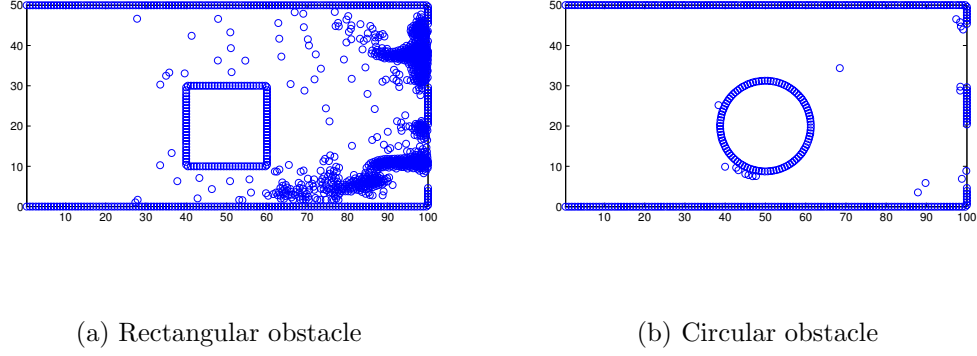


Figure 5.15: Comparison of circular and rectangular obstacles for  $t = 175s$  for relaxation time  $T = 0.001$ .

trance  $\{y = [35, 45], x = 0\}$  at the left boundary and leave the platform through the door  $\{y = [35, 45], x = 100\}$  at the right boundary. The density of the pedestrians increases from linearly from zero to a peak at  $t = 60sec$ , and then drops linearly back to zero at  $t = 120sec$ , and no pedestrians enter the platform thereafter. The density on inflow boundary is given by

$$\rho = \begin{cases} 2.5 \frac{t}{60}, & \text{if } 0 \leq t \leq 60 \\ -\frac{2.5}{60}(t - 120), & \text{if } 60 \leq t \leq 120 \\ 0, & \text{else} \end{cases} \quad (5.1.4)$$

We used other parameters same as in the square obstacle. We use a fixed random grid to solve the Eikonal equation. The grid distribution is given in the Figure 5.19. The pedestrian distribution along with the solution of the Eikonal equation for time  $t = 25s$ ,  $t = 50s$  and  $t = 100s$  are presented in Figures 5.20, 5.21 and 5.22, respectively. Though we do not have any benchmark to compare our results in this case, it is observed that the results from Figures 5.20, 5.21 and 5.22 appears to be following the pattern.

## 5.2 Numerical Example 2

In this section, we demonstrate the numerical example where the experimental results of pedestrian flow in a corridor and through T-junction are considered as our test case. We are interested to reproduce the experimental results of pedestrian flows in straight corridor [90, 89] and through T-junctions [91], i.e. our numerical results are compared to the experimental data.



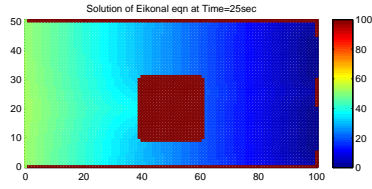
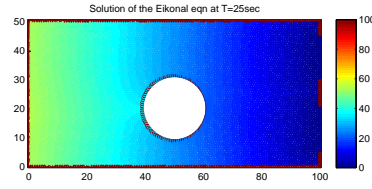
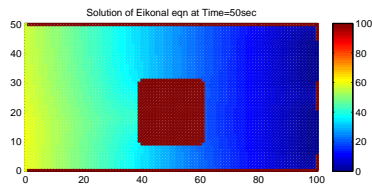
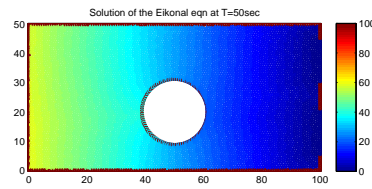
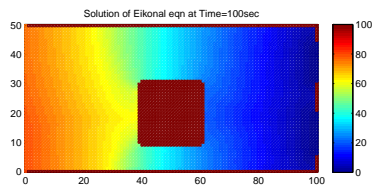
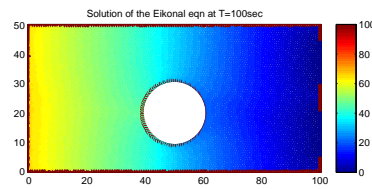
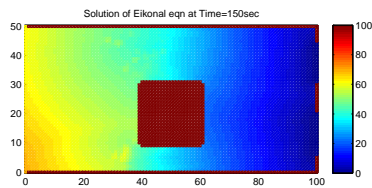
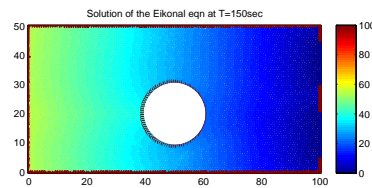
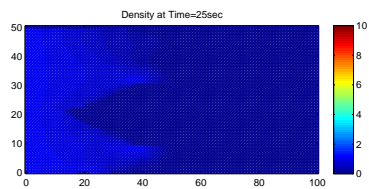
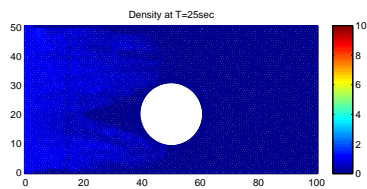
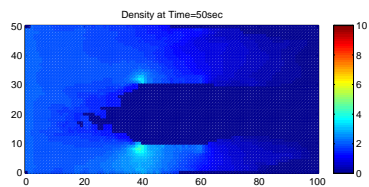
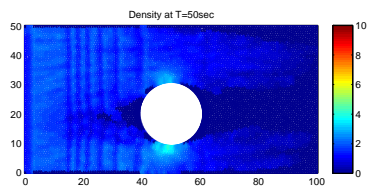
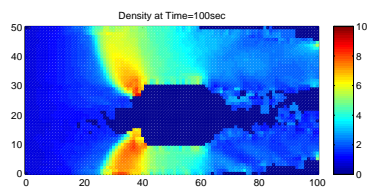
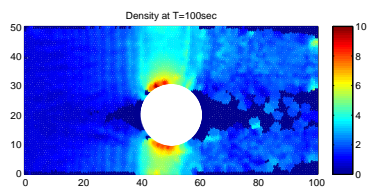
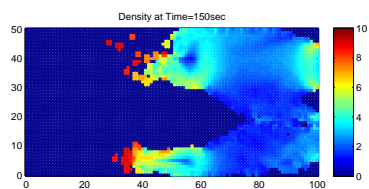
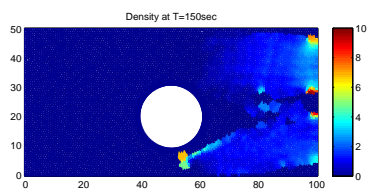
(a)  $t=25s$ (b)  $t=25s$ (c)  $t=50s$ (d)  $t=50s$ (e)  $t=100s$ (f)  $t=100s$ (g)  $t=150s$ (h)  $t=150s$ 

Figure 5.16: Solution of the Eikonal eqn for  $t = 25s$ ,  $t = 50s$ ,  $t = 100s$  and  $t = 150s$

(a)  $t=25s$ (b)  $t=25s$ (c)  $t=50s$ (d)  $t=50s$ (e)  $t=100s$ (f)  $t=100s$ (g)  $t=150s$ (h)  $t=150s$ Figure 5.17: Density for  $t = 25s$ ,  $t = 50s$ ,  $t = 100s$  and  $t = 150s$

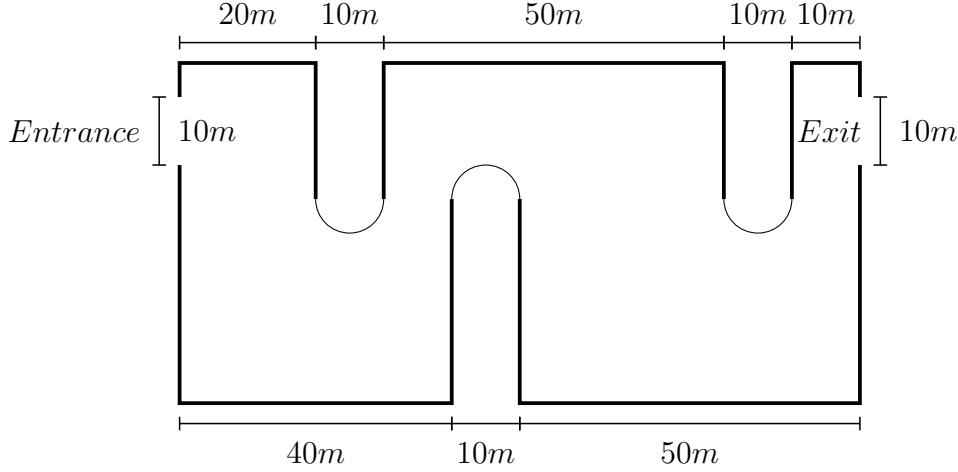


Figure 5.18: The railway platform with three obstacles.

We consider the Hydrodynamic equations (2.4.1, 2.4.3). For this example, we choose the following speed density relation[53]

$$U(\rho(x)) = u_{max} \left( 1 - \left( \frac{\rho}{\rho_{max}} \right)^{n_1} \right)^{n_2} \quad (5.2.1)$$

for some real values  $n_1$  and  $n_2$ , the maximum velocity  $u_{max}$  and the maximum density  $\rho_{max}$ .

### 5.2.1 Initial and Boundary Conditions

For our simulations, we consider the following initial conditions for the density and velocity

$$\rho_0(x) = \begin{cases} \rho_0, & \text{if } x \in \Omega_0 \\ 0, & \text{otherwise} \end{cases}$$

and

$$u = 0, \quad (5.2.2)$$

where  $\rho_0$  is a positive constant and  $\Omega_0$  describes a waiting area.

The boundary conditions for the Eikonal equation are

$$\begin{aligned} \Phi(x) &= 0, \text{ if } x \in \Omega_d \\ \Phi(x) &= \infty, \text{ if } x \in \Omega_w \end{aligned} \quad (5.2.3)$$

where  $\Omega_d$  is the destination for pedestrians and  $\Omega_w$  is the wall or obstacle in the domain.

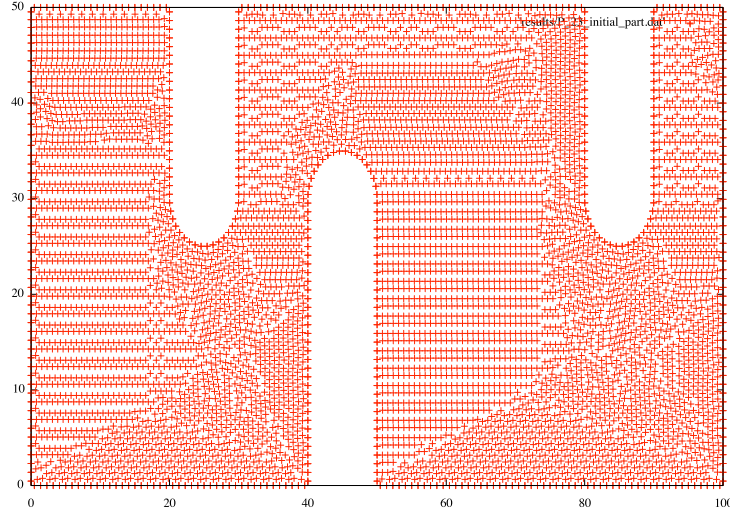


Figure 5.19: Randomly generated grid points.

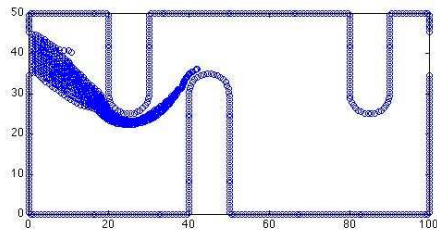
### 5.2.2 Parameters

We choose the desired speed and maximal density as in the experiments[90, 91]. The free flow velocity is  $1.55\text{ms}^{-1}$  and the maximum density is 3.7 pedestrians for  $\text{m}^2$ . Furthermore, we fix the time step size to be 0.0001 and set the relaxation parameter  $T$  as 0.001. The interaction constant  $k_n$  is 1000. The coefficient of friction  $\gamma_n$  and tangential coefficient  $\gamma_t$  are 10.0 and 2.0, respectively. The constants  $n_1$  and  $n_2$  in the speed-density relationship(5.2.1) are set to 0.4 and 0.8, respectively.

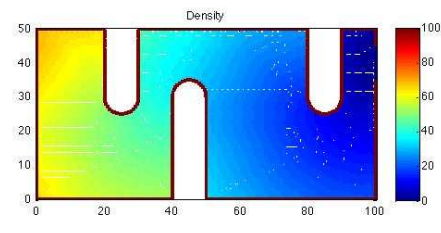
### 5.2.3 Results

#### Uni-directional pedestrian flow in a corridor

In the first case, we consider the uni-directional pedestrian flow in a straight corridor. The experimental set up from [89] is shown in Fig. 5.23. The corridor is of length  $8\text{m}$ . Initially, participants stay in the waiting area. There is a  $4\text{m}$  passage between the waiting area and the corridor to minimize the effect of  $b_{en}$ . In the experiments, the participants once they reach the exit of the corridor they will come back to the waiting area for another run. To emulate this behavior in our simulations, we provide the continuous stream of pedestrians. In Fig. 5.24, the trajectories of uni-directional pedestrian flow for high and low densities are shown. Fig.5.24(a) shows the trajectory

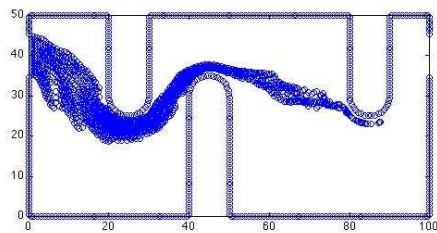


(a) Pedestrians distribution

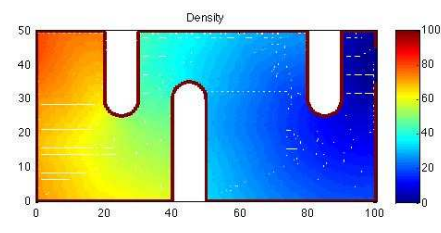


(b) Solution of the Eikonal equation

Figure 5.20: Distribution of pedestrians and solution of the Eikonal equation at  $t = 25s$ .



(a) Pedestrians distribution



(b) Solution of the Eikonal equation

Figure 5.21: Distribution of pedestrians and solution of the Eikonal equation at  $t = 50s$ .

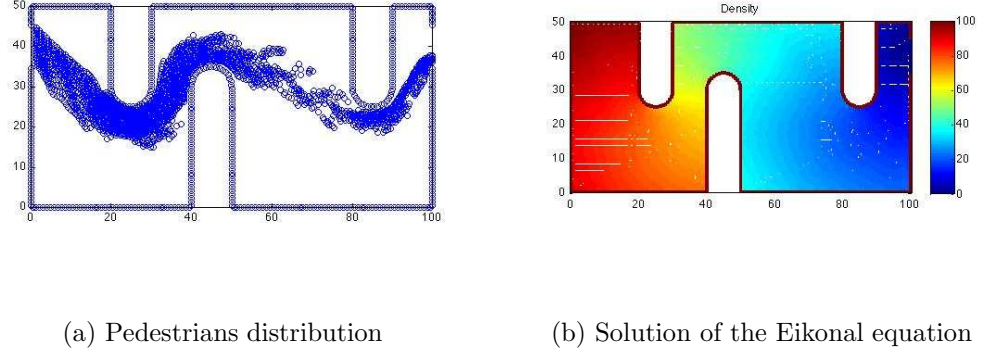


Figure 5.22: Distribution of pedestrians and solution of the Eikonal equation at  $t = 100s$ .

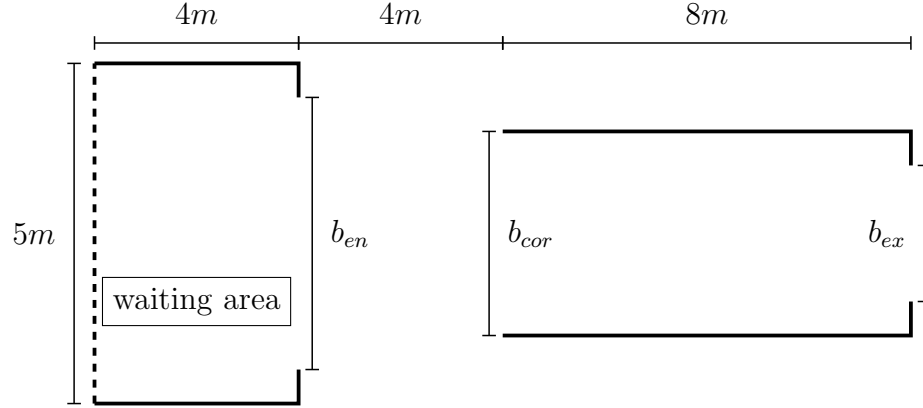


Figure 5.23: Straight corridor: sketch of the experimental setup where  $b_{cor}$  is the width of the corridor,  $b_{en}$  is the width of the entrance of the waiting area and  $b_{ex}$  is the width of the exit of the corridor.

of a uni-directional flow at high density and Fig. 5.24(b) shows the trajectory at low density. High density flow is obtained by  $b_{en} > b_{ex}$  and low density flow is obtained by  $b_{en} < b_{ex}$ . We investigate the influence of the corridor width on the fundamental diagram. Fig. 5.25 shows the relationship between density, velocity and flow for different sizes of corridor widths  $b_{cor} = 1.8m$ ,  $b_{cor} = 2.4m$  and  $b_{cor} = 3.0m$ . In Fig. 5.25 it can be observed that the width of the corridor has no effect on the fundamental diagram.

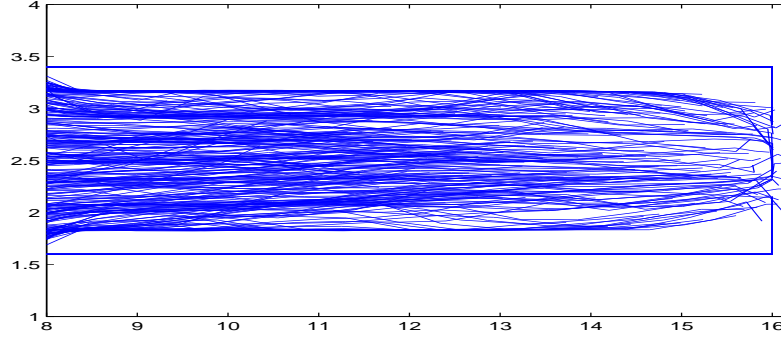
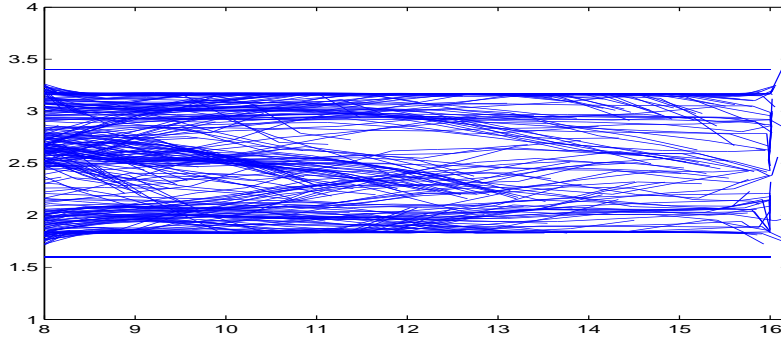
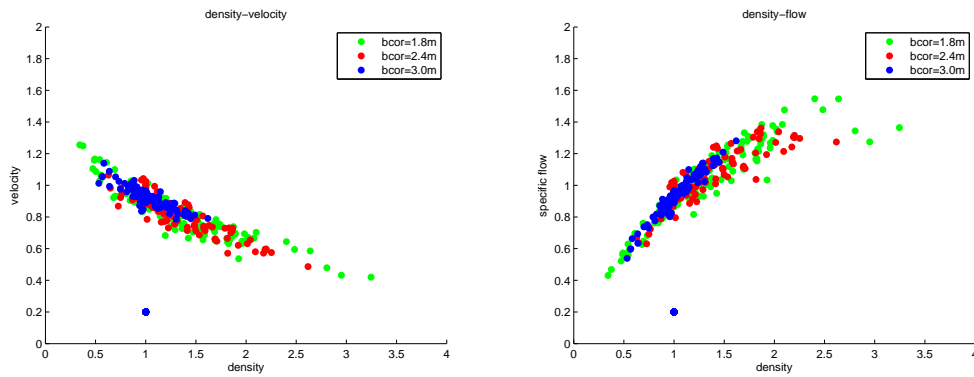
(a) High density:  $b_{en} > b_{ex}$ .(b) Low density:  $b_{en} < b_{ex}$ .

Figure 5.24: Trajectories of pedestrians inside the corridor.



(a) Fundamental diagram: density vs. velocity.

(b) Fundamental diagram: density vs. flow.

Figure 5.25: Comparison of fundamental diagrams for different corridor widths.

### Bidirectional pedestrian flow in Corridor

In the second case, we consider the bi-directional pedestrian flow in a straight corridor. The experimental set up from [90] is shown in Fig. 5.26.

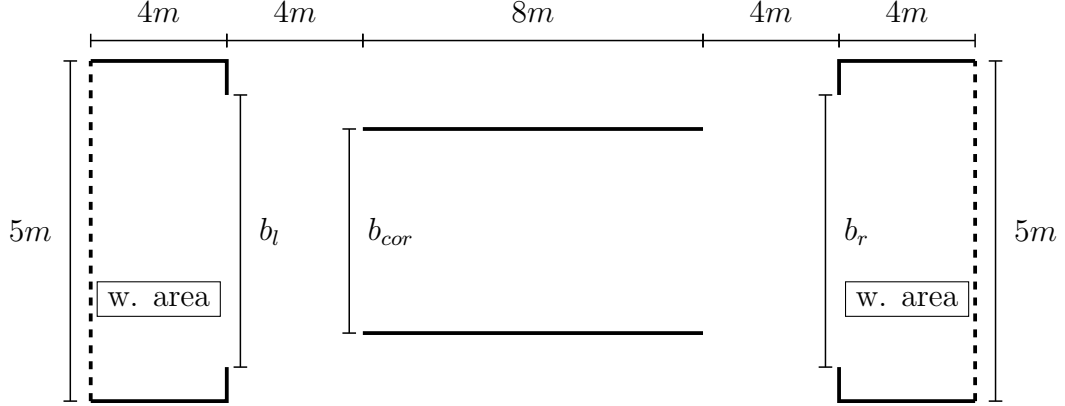


Figure 5.26: Straight corridor: sketch of the experimental setup where  $b_l$  and  $b_r$  are the width of the entrances of the waiting areas on the left and right side of the corridor.

The corridor is of length  $8m$  and  $b_{cor}$  is the width of the corridor. Initially, pedestrians stand in the waiting areas stationed at both sides of the corridor. There is a  $4m$  passage between the waiting area and the corridor to minimize the effect of  $b_l$  and  $b_r$ . In the experiments, when the participants arrive at the other side of the corridor, they leave the corridor and return to the waiting area for another run. In our simulations, we provide continuous stream of pedestrians in the waiting area to match with the experiments. To vary the form of the ordering, the participants get different instructions that result in different types of flows [90].

**BFR-SSL flow:** This type of flow is observed by using the same entrance width for both directions ( $b_l = b_r$ ) and giving no instructions to the participants about which exit they have to choose.

**BFR-DML flow:** In this case also  $b_l$  is same as  $b_r$ , but the instruction to the participants is changed. The participants were asked to choose an exit at the end of the corridor according to a number given to them in advance.

**UFR-DML flow:** In this case the widths of entrances  $b_l$  and  $b_r$  are different and the participants are instructed to choose an exit at the end of the corridor according to a number given at the beginning.

In our simulations, to emulate the BFR-SSL flow, we solve the Eikonal

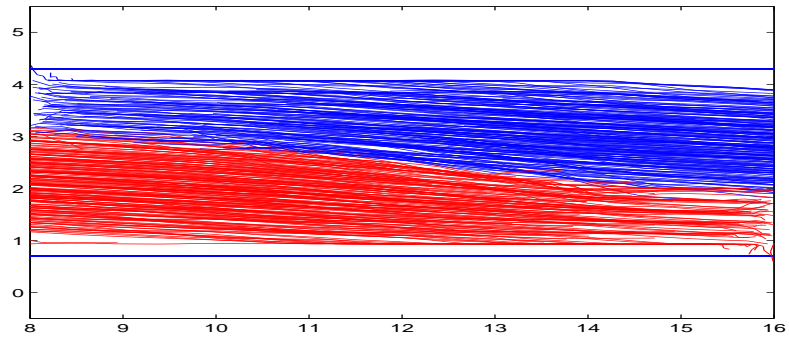


equation two times with different boundary conditions. Since for the pedestrians in left waiting area, the destination is the right end of the corridor and for pedestrians in right waiting area, the left end of the corridor is the destination. We solve the Eikonal equation with right and left ends of the corridor as boundary conditions and the corresponding Eikonal solutions are  $\Phi_1$  and  $\Phi_2$ , respectively. The velocity of left stream pedestrians is updated using  $\Phi_1$  and the velocity of right stream pedestrians is updated using  $\Phi_2$ . Similarly, to emulate the BFR-DML and UFR-DML flows, we solve the Eikonal equation four times with right-bottom, right-top, left-bottom and left-top corners of the corridor as boundary conditions and  $\Phi_1, \Phi_2, \Phi_3$  and  $\Phi_4$  present the corresponding solutions of the Eikonal equations. The velocity of odd-numbered left stream pedestrians will be updated by using  $\Phi_1$ , even-numbered left stream pedestrians will be updated using  $\Phi_2$ . The velocity of odd-numbered right stream velocity is updated by using  $\Phi_3$  and finally the velocity of even-numbered right stream pedestrians is updated by using  $\Phi_4$ . In Fig. 5.27, the trajectories of all these three flows are presented.

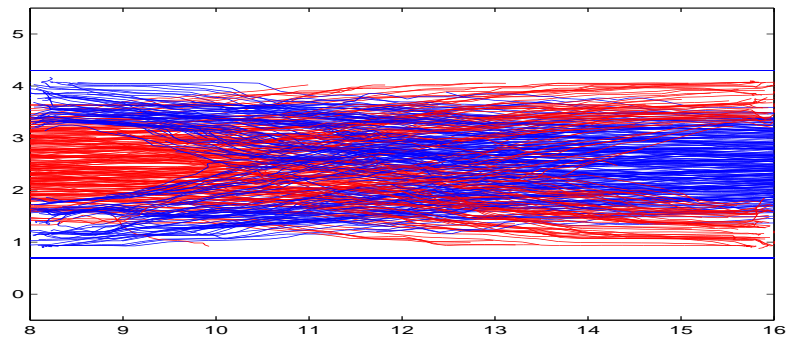
**Lane formation:** Lane formation is an important phenomenon in bi-directional flow. In Fig. 5.27, lane formation of pedestrians can be seen. In case of SSL flow, the pedestrians form two separate lanes and in case of DML flows, pedestrians form multiple lanes. In Figure 5.28, we plot the velocity profiles of BFR-DML flow for  $b_{cor} = 3.6m$  and  $b_l = b_r = 1.6m$  at  $t = 13sec$  and  $t = 51sec$ . We calculate the density in classical way. Our density profiles match with the experimental results [90].

**Fundamental diagram:** For the analysis of fundamental diagrams, a rectangle with a length of  $2m$  is chosen, see Fig. 5.26. To determine the fundamental diagram, we use the data from stationary flow. First, the influence of the corridor width on the fundamental diagram is studied. We consider  $b_{cor} = 3.6m$  and  $b_{cor} = 3.0m$  as two different widths of the corridor. Figure 5.29 shows the relationship between the density versus velocity and density versus specific flows. It can be observed that the fundamental diagrams provide good results compared to the experimental data.

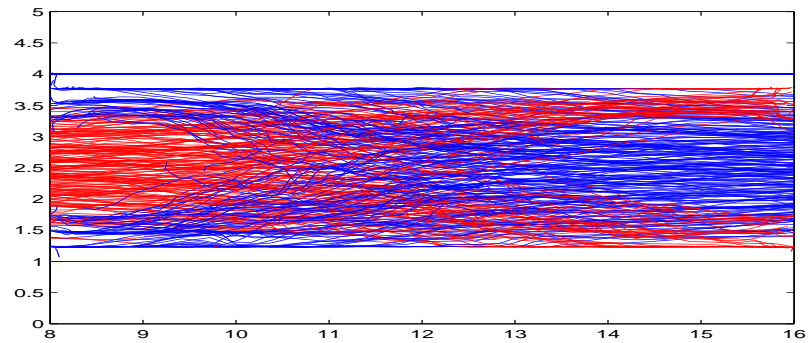
To investigate the influence of head-on conflicts and cross-directional conflicts in DML types of flow, numerical comparisons between the fundamental diagrams of SSL and DML flow for  $b_{cor} = 3.6m$  are performed. The comparisons are presented in Fig. 5.30 and obviously both are consistent with each other. This means, head-on conflicts in multilanes have the same influence on the fundamental diagram as conflicts at the borders in stable separated lane flow. Due to limited computational time resources, the fundamental diagram is computed only for density values less than  $2.0m^{-2}$  in the experiments for  $b_{cor} = 3.0$ .



(a) BFR-SSL flow.



(b) BFR-DML flow.



(c) UFR-DML flow.

Figure 5.27: Trajectories of pedestrians inside the corridor.

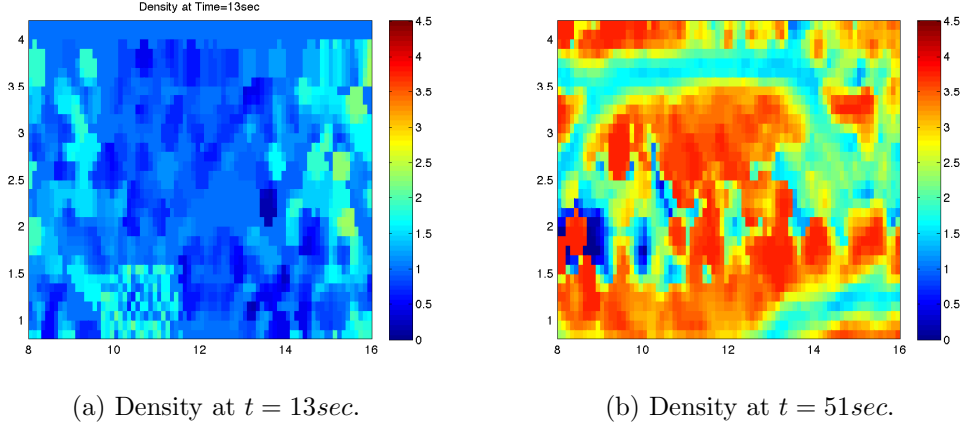
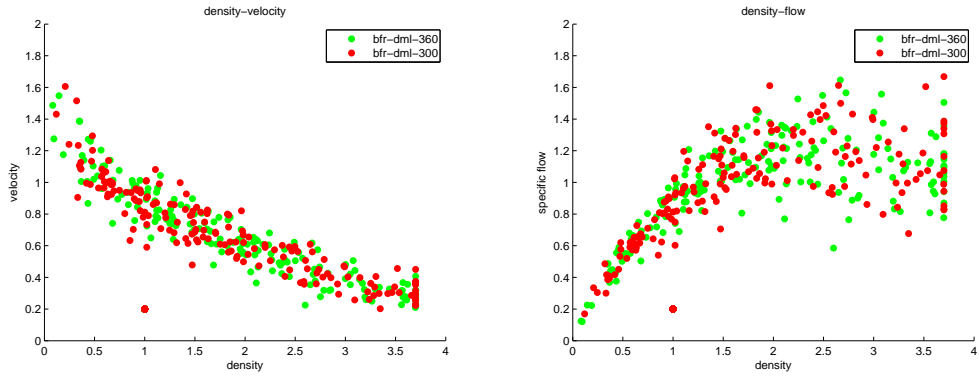
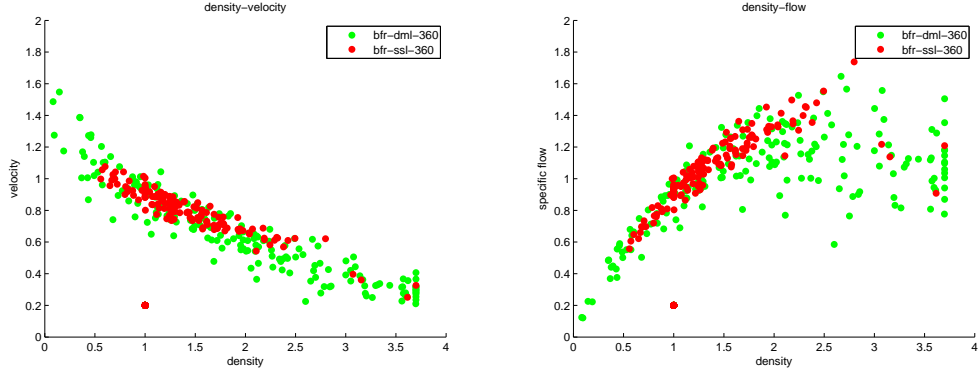
Figure 5.28: Density profiles at time  $t = 13$  and  $t = 51$  seconds.

Figure 5.29: Comparison of fundamental diagrams of DML flow for different corridor widths.



(a) Fundamental diagram: density vs. velocity. (b) Fundamental diagram: density vs. flow.

Figure 5.30: Comparison of fundamental diagrams for SSL and DML flow.

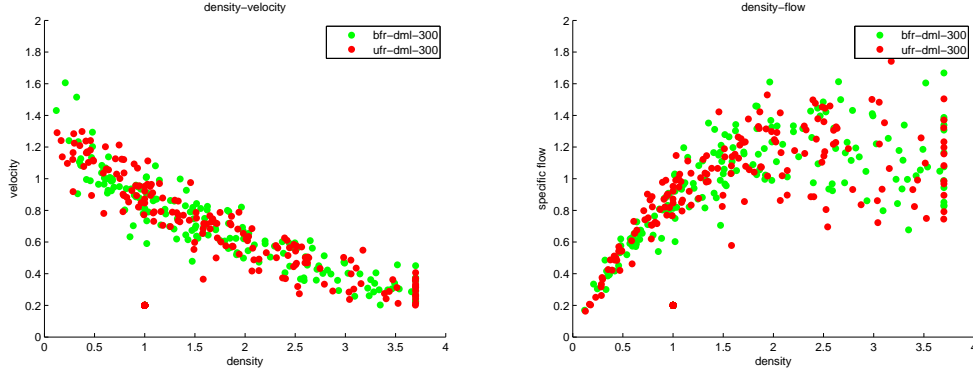
Finally, the influence of flow ratio of opposing streams on the fundamental diagram is studied. We compare the fundamental diagrams of BFR and UFR flow in Figure 5.31. It can be seen that the asymmetry in the flows does not affect the fundamental diagrams. In all three cases, our numerical results are consistent with the experimental data.

### Pedestrian flow in T-junction

In the third case, the pedestrian flow through a T-junction is considered. The experimental set up for the T-junction [91] is shown in Fig. 5.32.

Initially, the pedestrians are located in the waiting areas stationed at the left and right sides of the T-junction. Pedestrians move from two branches oppositely and then merge into the main stream at the T-junction. Here,  $b_{cor1}$  is the width of the corridor where pedestrians enter and  $b_{cor2}$  is the width of the corridor where pedestrians exit the T-junction, as shown in Fig. 5.32. To emulate the experiments, we provide the continuous pedestrian streams from both directions. As in the corridor example, there is a 4m passage between the T-junction and the waiting areas to minimize the effect of entrance. In this way, the flow in the corridor was nearly homogeneous over its entire width. Fig. 5.33 shows the pedestrian trajectories through the T-junction.

In Fig 5.34, density profiles for low density and high density situations are presented. Density of the flow is varied by changing the width of the entrance of the waiting areas. For low density flow situation, the widths of the waiting



(a) Fundamental diagram: density vs. velocity. (b) Fundamental diagram: density vs. flow.

Figure 5.31: Comparison of fundamental diagrams for BFR and UFR flow.

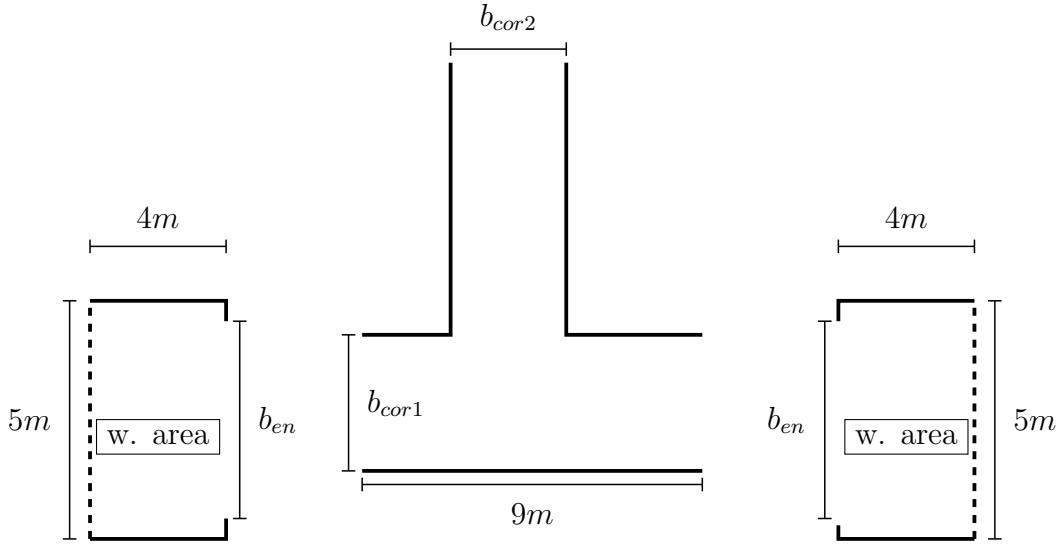


Figure 5.32: T-junction: sketch of the experimental setup where  $b_{en}$  is the width of the entrances of waiting areas at the left and right sides of the T-junction.

area are set as  $b_{en} = 2.4m$  and for high density flow situation, the widths of the waiting area are set as  $b_{en} = 0.5m$ . The density distribution in T-junction is not homogeneous both for low and high density situations. For low density situation, the higher density region locates at the main stream after merging

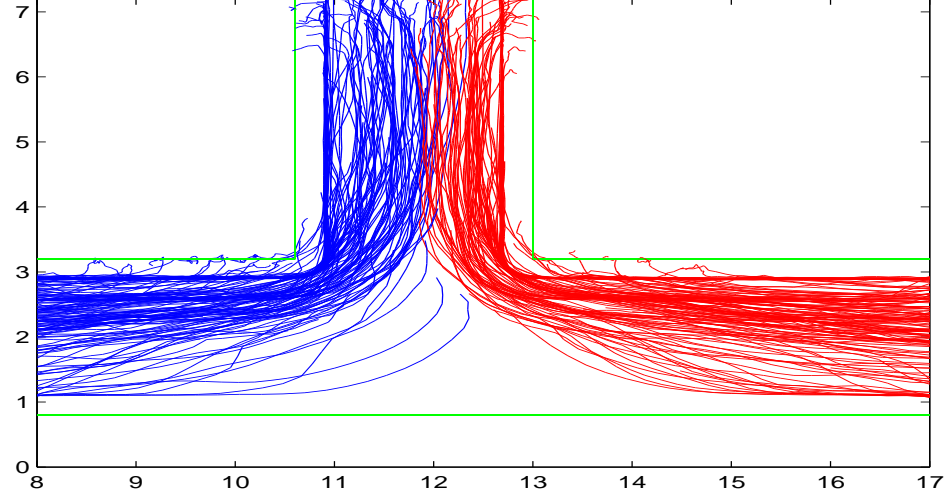


Figure 5.33: Trajectory of pedestrians through a T-junction.

and for the high density situation, high density region locates at junction. In Fig. 5.35, we compare the fundamental diagrams. The data assigned with 'T-left' and 'T-right' are measured in the areas before the streams merge, while the data assigned with 'T-front' are measured in the region where the streams have already merged. The locations of these measurements can be seen in the Fig. 5.32. In the experimental results by [91], the velocities of the pedestrians after merging is higher than the velocities of pedestrian before merging. This is because, before the merging pedestrians slow down near the corner when approaching pedestrians from the opposite stream. Another reason might be the fact that, when the destination is visible, pedestrians tend to walk fast. In our model, this idea is not included and can be considered as future work.

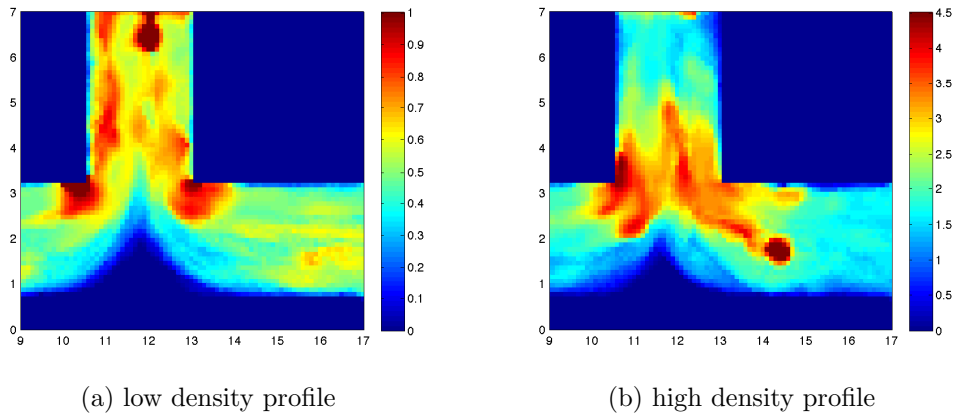


Figure 5.34: Density profiles for low and high density situations.

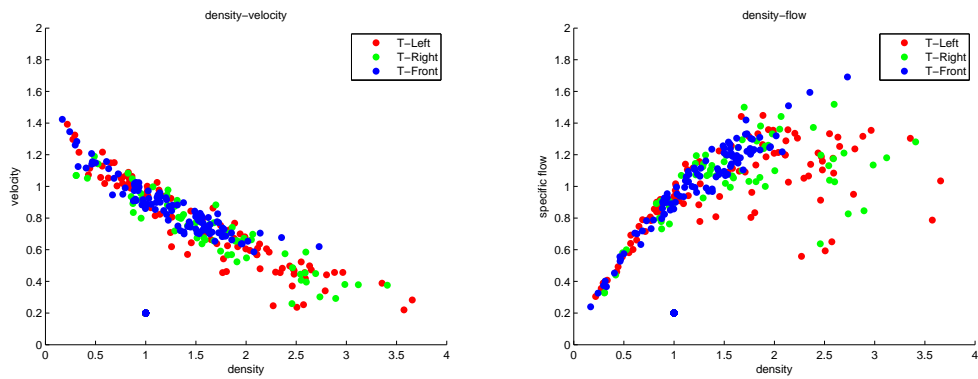


Figure 5.35: Fundamental diagrams of pedestrian flow at different measurement locations at a T-junction.





# Chapter 6

## Conclusions

Pedestrian models have variety of applications in many areas. In this thesis a hierarchy of models for pedestrian flow from microscopic to non-local continuum models is presented. We started with the microscopic model for pedestrian flow and then proceeded to derive the hydrodynamic as well as scalar models from the microscopic model. Optimal path for pedestrians is computed using the solution of the non linear Eikonal equation. The particle methods we used to solve the governing equations are presented.

The new mesh free method we presented to solve the Eikonal equation has got the advantage that it is applicable for any arbitrary grid and on complex geometries. The validation of our method is done by presenting some numerical examples taken from literature. We have discussed these examples for both structured and random grid and it is observed that our method is converging for both kinds of grids.

We have presented two numerical examples in pedestrian modelling. In our first numerical example, we consider the pedestrian flow in a railway platform with an obstacle in the middle. We compared microscopic, hydrodynamic and scalar models for this configuration and these three models are in good agreement. We analysed this configuration for different parameters such as relaxation time, interaction coefficient. We have also analysed pedestrian flow for different types of obstacles such as square, circle and multiple obstacles.

In the second numerical example, we compared our results with the experimental results. We consider experimental results of pedestrian flow in a straight corridor as well as through T-junction. In straight corridor case,

we studied the lane formation, fundamental diagrams for different situations in unidirectional and bidirectional pedestrian flow and compared our results with experimental results. In the pedestrian flow through T-junction situation, we have studied the fundamental diagrams for various cases. In all these cases, our simulation results are in good agreement with the experimental results.

# List of Figures

1.1	Pedestrian flow in public buildings; Source : (a) knoxnews.com (b) scabee.com . . . . .	2
1.2	Pedestrians on streets; Source: (a) chicagotribune.com (b) farmeronamission.blogspot.com . . . . .	3
1.3	Crowd gatherings; Source : (a) ramadan-karim.com (b) rnw.nl	5
2.1	Additional repulsive force on magnetic force model. . . . .	12
3.1	Grid stencil. . . . .	34
3.2	Beginning of the Fast marching method. . . . .	35
3.3	Domain. . . . .	39
3.4	Randomly generated grid points . . . . .	47
3.5	Solution description of Example 1 . . . . .	49
3.6	Solution description of Example 2 . . . . .	50
3.7	Solution description of Example 3 . . . . .	52
3.8	Computational Domain . . . . .	53
3.9	Path Planning . . . . .	54
4.1	Flow domain with non-structured grid . . . . .	58
5.1	The railway platform with a square obstruction. .	64
5.2	The railway platform with a circular obstruction.	64
5.3	Density at $t = 120$ along $x = 36$ and $y = 33$ for hydrodynamic, scalar and microscopic models. . . . .	68
5.4	Density of hydrodynamic and scalar models at $t = 120s$ . . . .	69

5.5	Distribution of particles for microscopic, hydrodynamic and simplified models at $t = 25s$ . . . . .	70
5.6	Distribution of particles for microscopic, hydrodynamic and simplified models at $t = 50s$ . . . . .	71
5.7	Distribution of particles for microscopic, hydrodynamic and simplified models at $t = 100s$ . . . . .	72
5.8	Distribution of particles for microscopic, hydrodynamic and simplified models at $t = 150s$ . . . . .	73
5.9	Solution of the Eikonal equation and density for $t = 25s$ , $t = 50s$ , $t = 100s$ and $t = 150s$ . . . . .	74
5.10	Distribution of particles for hydrodynamic and scalar model at $t = 100s$ for relaxation time $T = 0.001$ . . . . .	75
5.11	Distribution of particles for hydrodynamic and scalar model at $t = 100s$ for relaxation time $T = 0.01$ . . . . .	76
5.12	Distribution of particles for hydrodynamic and scalar model at $t = 100s$ for relaxation time $T = 0.1$ . . . . .	76
5.13	Distribution of particles for hydrodynamic and scalar model at $t = 100s$ for relaxation time $T = 1.0$ . . . . .	77
5.14	Comparison of circular and rectangular obstacles for $t = 150s$ for relaxation time $T = 0.001$ . . . . .	77
5.15	Comparison of circular and rectangular obstacles for $t = 175s$ for relaxation time $T = 0.001$ . . . . .	78
5.16	Solution of the Eikonal eqn for $t = 25s$ , $t = 50s$ , $t = 100s$ and $t = 150s$ . . . . .	79
5.17	Density for $t = 25s$ , $t = 50s$ , $t = 100s$ and $t = 150s$ . . . . .	80
5.18	The railway platform with three obstacles. . . . .	81
5.19	Randomly generated grid points. . . . .	82
5.20	Distribution of pedestrians and solution of the Eikonal equation at $t = 25s$ . . . . .	83
5.21	Distribution of pedestrians and solution of the Eikonal equation at $t = 50s$ . . . . .	83
5.22	Distribution of pedestrians and solution of the Eikonal equation at $t = 100s$ . . . . .	84

5.23	Straight corridor: sketch of the experimental setup where $b_{cor}$ is the width of the corridor, $b_{en}$ is the width of the entrance of the waiting area and $b_{ex}$ is the width of the exit of the corridor.	84
5.24	Trajectories of pedestrians inside the corridor. . . . .	85
5.25	Comparison of fundamental diagrams for different corridor widths. . . . .	85
5.26	Straight corridor: sketch of the experimental setup where $b_l$ and $b_r$ are the width of the entrances of the waiting areas on the left and right side of the corridor. . . . .	86
5.27	Trajectories of pedestrians inside the corridor. . . . .	88
5.28	Density profiles at time $t = 13$ and $t = 51$ seconds. . . . .	89
5.29	Comparison of fundamental diagrams of DML flow for different corridor widths. . . . .	89
5.30	Comparison of fundamental diagrams for SSL and DML flow.	90
5.31	Comparison of fundamental diagrams for BFR and UFR flow.	91
5.32	T-junction: sketch of the experimental setup where $b_{en}$ is the width of the entrances of waiting areas at the left and right sides of the T-junction. . . . .	91
5.33	Trajectory of pedestrians through a T-junction. . . . .	92
5.34	Density profiles for low and high density situations. . . . .	93
5.35	Fundamental diagrams of pedestrian flow at different measurement locations at a T-junction. . . . .	93



# List of Tables

1.1	Crowd disasters; Source : <a href="http://en.wikipedia.org/wiki/List-of-human-stampedes">en.wikipedia.org/wiki/List-of-human-stampedes</a> . . . . .	4
3.1	Error values for Example 1: Cartesian Grid . . . . .	48
3.2	Error values for Example 1: Unstructured Grid . . . . .	48
3.3	Error values for Example 2: Cartesian Grid . . . . .	51
3.4	Error values for Example 2: Unstructured Grid . . . . .	51
3.5	Error values for Example 3: Cartesian Grid . . . . .	51
3.6	Error values for Example 3: Unstructured Grid . . . . .	53
5.1	Parameters . . . . .	66
5.2	Microscopic model: Particles per cell $N_{max}$ , computing times for $t = 150s$ and maximal total number of pedestrians. . . . .	66
5.3	Hydrodynamic and scalar model: Computing times for $t = 150s$ for $dx = 1.0$ and $dx = 0.7143$ . . . . .	67





# Bibliography

- [1] D. Adalsteinsson and J. Sethian, A Fast level Set Method for propagating Interfaces, *J. Computational Physics*, vol 118, pp.269-277,1995.
- [2] D. Amadoria and M. Di Francesco, The one-dimensional Hughes model for pedestrian flow: Riemann-type solutions, *Acta Mathematica Scientia*, vol 32,no 1, 2012, pp. 259-280.
- [3] M.Bardi and I.Capuzzo-Dolcetta, *Optimal control and viscosity solutions of Hamilton-Jacobi equations*, Birkhauser, Boston,1997.
- [4] N. Bellomo and C. Dogbe, On the Modeling of Traffic and Crowds: A Survey of Models, Speculations, and Perspectives, *Siam Review*, 53/3, (2011), pp. 409-463.
- [5] N. Bellomo and C. Dogbe, On the modelling crowd dynamics from scaling to hyperbolic macroscopic models, *Math. Models Methods Appl. Sci.*, 18 (Suppl.) (2008), pp. 1317-1345.
- [6] V.J. Blue and J.L. Adler., Emergent Fundamental Pedestrian Flows from Cellular Automata Microsimulation, *Transportation Research Record*,1644,29-36, 1998.
- [7] V.J. Blue and J.L. Adler., Using Cellular Autometa. Microsimulation to Model Pedestrian Movements, *Proceedings of the 14th International Symposium on Transportation and Traffic Theory*, A. Ceder (ed.) Elsevier Science Ltd, pp. 135-154,1999.
- [8] V.J. Blue and J.L. Adler., Cellular Autometa Microsimulation of Bidirectional Pedestrian Flows, *Transportation Research Board*, 1678, pp 135-141, 2000.
- [9] W. Braun and K. Hepp, The Vlasov Dynamics and Its Fluctuations in the  $1/N$  Limit of Interacting Classical Particles, *Commun. Math. Phys*, **56** (1977), pp. 101-113.
- [10] J.A. Cañizo, J.A. Carrillo, and J. Rosado, A well-posedness theory in measures for some kinetic models of collective motion, *Math. Mod. Meth. Appl. Sci.*, 21, (2011), pp. 515-539.

- [11] J.A. Carrillo, M.R. D’Orsogna and V. Panferov, Double milling in self-propelled swarms from kinetic theory. *Kinetic and Related Models*, vol 2, no 2, (2009), pp. 363-378.
- [12] J. A. Carrillo, A. Klar, S. Martin and S. Tiwari, Self-propelled interacting particle systems with roosting force, *Math. Mod. Meth. Appl. Sci.*, 20, (2010), pp. 1533-1552.
- [13] Y.Cheng and C.W.Shu, A discontinuous Galerkin finite element method for directly solving Hamilton-jacobi equations, *J.Comput.Phys*, 223:398-415 (2007).
- [14] D.L. Chopp, Computing minimal surfaces via level set curvature flow, *Jour. of. Comp. Phys.*, 106, pp. 77-91, 1993.
- [15] Y.L. Chuang, M.R. D’Orsogna, D. Marthaler, A.L. Bertozzi and L. Chayes, State transitions and the continuum limit for a 2D interacting, self-propelled particle system. *Physics D*, 232 (2007), pp. 33-47.
- [16] R. Colombo, M. Garavello, and M. Lecureux-Mercier, A Class of Non-Local Models for Pedestrian Traffic, *MMMAS*, 22 (4), 1150023 , 2012.
- [17] R. Colombo, M. Garavello, and M. Lecureux-Mercier, Non -local crowd dynamics, *Comptes Rendus Mathematique*, vol 349, no 13, pp. 769-772, 2011.
- [18] R.M. Colombo and M.D. Rosini, Pedestrian flows and non-classical shocks, *Math. Methods Appl. Sci.*, 28 (13) (2005), pp. 1553-1567.
- [19] J.F. Coulombel, F. Golse and T. Goudon, Diffusion approximation and entropy based moment closure for kinetic equations, *Asymptotic Analysis*, 45 (1-2), 1-39, 2005.
- [20] M.G. Crandall and P.L.Lions, Viscosity solutions of Hamilton-Jacobi equations, *Trans. Amer. Math. Soc*, 1983.
- [21] P. Degond, C. Appert-Rolland, M. Moussaid, J. Pettre and G. Theraulaz, A hierarchy of heuristic-based models of crowd dynamics, *Journal of Statistical Physics*, vol 152, no 6, pp 1033-1068, 2013.
- [22] P. Degond and S. Motsch, Continuum limit of self-driven particles with orientation interaction. *Math. Models Methods Appl. Sci*, 18 (2008), pp. 1193-1215.
- [23] P. Degond and F.J. Mustieles, Approximation of diffusion equations by deterministic convections of particles, *SIAM J. on Scientific and Statistical Computing* 11, (1990), pp. 293-310.
- [24] G. A. Dilts, Moving least squares particle hydrodynamics I, consistency and stability, *Hydrodynamics methods group report, Los Alamos National Laboratory* (1996).
- [25] C. Dogbe, On the modelling of crowd dynamics by generalized kinetic models, *J.Math.Anal. Appl*, 387, 512-532, 2012.

- [26] M. Dehghan and R.Salehi, A boundary-only mesh less method for numerical solution of the Eikonal equation, *Compt. Mech*, (2011) vol 47, pp. 283-294.
- [27] T. Deschamps and L.D. Cohen, Fast Extraction of Tubular and tree 3D surfaces with Front propagation methods, *Proc. Int'l Conf. Pattern Recognition*, PP. 473-476,2005.
- [28] R. Etikyala, S. Göttlich, A. Klar and S. Tiwari, Particle methods for pedestrian flow models: from microscopic to non-local continuum models, to be appear in *Mathematical Models and Methods in Applied Sciences*, Vol. 24(10), 2014.
- [29] R. Etikyala, S. Göttlich, A. Klar and S. Tiwari, A macroscopic model for pedestrian flow: comparisons with experimental results of pedestrian flow in corridors and T-junctions, submitted to *Neural, Parallel and Scientific Computations [CNLS-2013]*.
- [30] M.Falcone and R.ferretti, Semi-Lagrangian schemes for Hamilton-Jacobi equations, discrete representation formulae and Godunov methods, *J.Computational Phy*, 175 (2002),559-575.
- [31] M. Di Francesco, P.A. Markowich, J.F. Pietschmann and M.T. Wolfram, On the Hughes' model for pedestrian flow: The one-dimensional case, *J. Differential Equations* 250 (2011), pp. 1334-1362.
- [32] P.G.Gipps and B. Marksjo, A Micro-simulation Model for Pedestrian Flows, *Mathematics and Computers in Simulation*, 27, pp.95-105,1985.
- [33] P.A. Gremaud and C.M. Kuster, Computational study of fast methods for the Eikonal equations, *SIAM J. Sci. Comput*, 27 (2006) 1803-1816.
- [34] S.Y. Ha and E. Tadmor, From particle to kinetic and hydrodynamic descriptions of flocking. *Kinetic and Related Models*, 1 (2008), pp. 415-435.
- [35] D.Helbing, A Mathematical model for behavior of pedestrians, *Behavioral Science*, 36,pp. 198-310, 1991.
- [36] D. Helbing, A fluid dynamic model for the movement of pedestrians, *Complex Systems*, 6, 391-415,1992.
- [37] D. Helbing and P. Molnar, Social force model for pedestrian dynamics, *Phys. Rev. E*, 51 (1995), pp. 4282-4286.
- [38] D. Helbing, and T. Vicsek, Optimal Self-Organization, *New Journal of Physics* 1, 13.1-13.17,1999.
- [39] D. Helbing, Traffic and related self-driven many-particle systems, *Rev. Modern Phys*, 73 (4) (2001),pp. 1067-1141.
- [40] D. Helbing, I.J. Farkas, P. Molnar and T. Vicsek, Simulation of pedestrian crowds in normal and evacuation situations, in: M. Schreckenberg, S.D. Sharma (Eds.), *Pedestrian and Evacuation Dynamics*, Springer-Verlag, Berlin, 2002, pp. 21-58.

- [41] L.F.Henderson, On the Fluid Mechanisc of Human Crowd Motions, *Transportation Research* 8, pp. 509-515,1974.
- [42] B.Horn, *Robot Vision*, MIT Press,1986.
- [43] R.L. Hughes, A continuum theory for the flow of pedestrians,*Transp. Res. Part B: Methodological* 36 (6) (2002), pp. 507-535.
- [44] R.L. Hughes, The flow of human crowds, *Annu. Rev. Fluid Mech*, 35 (2003), pp. 169-182.
- [45] R.Isaacs, *Differential games*, wiley, New York, 1965.
- [46] G.-S. Jiang and D.Peng, Weighted ENO Schemes for Hamilton-Jacobi equations, *SIAM Journal on Scientific Computing*, vol 21(2000), pp.2126-2143.
- [47] C. Kao, S. Osher and R.Tsai, Fast Sweeping Methods for Hamilton-jacobi Equations, *SIAM J. Numerical Analysis*, vol 42, pp.2612-2632, 2005.
- [48] S.Kim, ENO-DNO-PS: A Stable, Second-Order Accuracy Eikonal Solver, *Soc. Exploration Geophysicists*, pp. 1747-1750, 1999.
- [49] R.Kimmel and J. A. Sethian, Optimal Algorithm for shape from Shading and Path Planning, *J.Computational Physics*, vol 118, pp.269-277, 1995.
- [50] R.Kimmel and J.Sethian, Fast Marching Methods on Triangulated Domains, *Proc. Nat'l Academy of Sciences*, vol 95, no. 11, pp. 8341-8435, 1998.
- [51] A. Klar and S. Tiwari, Underresolved meshfree methods for hydrodynamic interacting particle models preprint
- [52] A.Klar, S.Tiwari and E. Raghavender, Mesh Free method for Numerical Solution of The Eikonal Equation, Proceedings of International workshop on PDE Modeling and Computation, *Advances in PDE Modeling and Computation*, Ane Books Pvt. Ltd, 2013.
- [53] R.D. Kuhne and M.B. Rodinger, Macroscopic simulation model for freeway traffic with jams and stop-start waves. In: Nelson, B.L., Kelton, W.D., Clark, G.M. (eds): Proceedings of the 1991 Winter Simulation Conference, Pheonix, AZ, p. 762-770 (1991).
- [54] J. Kuhnert, General smoothed particle hydrodynamics, *PhD Thesis, University of Kiaserslautern, Germany* (1999).
- [55] F. Li, CW. Shu, YT.Zhao and H.Zhao, A second order discontinuous Galerkin fast sweeping method for Eikonal equations, *J.Comput. Phys* 227:8191-8208 (2008).
- [56] H. Ling, S.C. Wong, M. Zhang, C.-H. Shu and W.H.K. Lam, Revisiting Hughes dynamic continuum model for pedestrian flow and the development of an efficient solution algorithm, *Transp. Res. Part B: Methodological*, 43 (1) (2009), pp. 127-141.

- [57] C. D. Levermore, Relating Eddington factors to flux limiters, *J. Quant. Spectrosc. Radiat. Transf.*, 31, 149-160, 1984.
- [58] G.G. Lovas, Modeling And Simulation of Pedestrian Traffic Flow, *Transportation Research 28B*, pp. 429-443, 1994.
- [59] B. Maury, A. Roudneff-Chupin and F. Santambrogio, A macroscopic crowd motion model of the gradient-flow type, *Mathematical Models and Methods in Applied Sciences*, Vol. 20, No. 10 (2010), pp. 1787-1821.
- [60] A. Mogilner, L. Edelstein-Keshet, L. Bent, and A. Spiros, Mutual interactions, potentials, and individual distance in a social aggregation, *J. Math. Biol.*, vol 47, (2003), pp. 353-389.
- [61] M. Muramatsu, T. Irie, and T. Nagatani, Jamming Transition In Pedestrian Counter Flow, *Physica A: Statistical Mechanics And Its Applications*, 267 pp. 487-498, 1999.
- [62] H. Neunzert, The Vlasov equation as a limit of Hamiltonian classical mechanical systems of interacting particles, *Trans. Fluid Dynamics*, vol 18 (1977), pp. 663-678.
- [63] S. Okazaki, A Study of Pedestrian Movement in Architectural Space, Part 1: Pedestrian Movement by the Application on of Magnetic Models. *Trans. of A.I.J.*, vol 35, No. 283, pp. 111-119, 1979.
- [64] S. J. Osher and R. Fedkiw, *Level set methods and dynamic implicit surface*, Springer, 2003.
- [65] S. Osher and J.A. Sethian, Fronts propagating with curvature dependent speed: algorithms based on Hamilton-Jacobi formulations, *J. Computational Phy*, 79 (1988) 12-49.
- [66] S. Osher, A level set formulation for the solution of the Dirichlet problem for Hamilton-Jacobi equations, *SIAM Journal on Mathematical Analysis*, 24(1993), 1145-1152.
- [67] B. Piccoli and A. Tosin, Pedestrian flows in bounded domains with obstacles, *Contin. Mech. Thermodyn*, 21 (2) (2009) 85-107.
- [68] P. Podvin and I. Lecomte, Finite Difference Computation of Travel times in Very Contrasted velocity Models: A Massively Parallel Approach and Its Associated Tools, *Geophysical J. Int'l*, vol 105, pp. 271-284, 1991.
- [69] J. Qian and W.W. Symes, Finite-difference quasi-P traveltimes for anisotropic media, *Geophysics*, vol 67, no 1, (2002), 147-155.
- [70] F. Qin and G.T. Schuster, First-arrival traveltime calculation for anisotropic media, *Geophysics*, 58 (1993), 1349-1358.
- [71] E. Rouy and A. Tourin, A viscosity solutions approach to shape from shading, *SIAM J. Num. Anal.*, vol 29, no 3, pp. 867-884, 1992.

- [72] J.A. Sethian, *Level set methods*, Cambridge Univ. Press, 1996.
- [73] J. Sethian, *Level Sets Methods and Fast Marching Methods*, second ed. Cambridge Univ. Press, 1999.
- [74] J.A.Sethian and A.Vladimirsky, Fast Methods for the Eikonal and Related Hamilton-Jacobi Equations on Unstructured Meshes, *pro. nat'l Academy of Sciences*, vol 97, no 11, pp. 5699-5703, 2000.
- [75] J.A.Sethian and A.Vladimirsky, Ordered upwind methods for static Hamilton-Jacobi equations: theory and algorithms, *SIAM Journal on Numerical Analysis*, 2003, 41(1), pp. 325-363.
- [76] J. A. Sethian, Fast Marching Methods, *SIAM Review*, 41, 1999, 199-235.
- [77] Lee Soomaroo and Virginia Murray, Disasters at Mass Gatherings: Lessons from History, PLOS Currents, 2012.
- [78] H. Spohn, Large scale dynamics of interacting particles. *Texts and Monographs in Physics*, Springer (1991).
- [79] P.A. Thompson, and E.W. Marchant, Testing and Application of the Computer Model "SIMULEX", *Fire Safety Journal* 24, pp. 149-166, 1995.
- [80] P.A. Thompson, and E.W. Marchant, A Computer Model the Evacuation of Large Building Populations, *Fire Safety Journal* 24, pp. 131-148, 1995.
- [81] S.Tiwari and J.Kuhnert, Modeling of two-phase flow with surface tension by finite pointset method (FPM), *J. Comp. Appl. Math*, 203 (2007), pp. 376-386.
- [82] S.Tiwari and J.Kuhnert, Grid free method for solving Poisson equation, preprint, *berichte des fraunhofer ITWM*, kaiserslautern, Germany, Nr. 25 (2011).
- [83] A. Treuille, S. Cooper and Z. Popovic, Continuum crowds, in: ACM Transaction on Graphics, *Proceedings of SCM SIGGRAPH*, 2006, vol. 25, pp. 1160 - 1168.
- [84] J.N. Tsitsiklis, Efficient algorithms for globally optimal trajectories, *IEEE Tran. Automatic Control*, 40:1528-1538, 1995.
- [85] J.Vidale, Finite-Difference Calculation of Traveltimes in three Dimensions, *J.Geophysics*, vol 56, pp 812-821, 1991.
- [86] J.M.Watts, Computer Models for Evacuation Analysis, *Fire Safety Journal*, 12 pp. 237-245, 1987.
- [87] Y.T.Zhang and H.K Zhao and J.Qian, Higher Order Fast Sweeping Methods for Static Hamilton-Jacobi Equations, *Journal of Scientific Computing*, vol 29, no 1, pp. 25-56, 2006.
- [88] Y.T. Zhang and C.W. Shu, Higher order WENO schemes for Hamilton-Jacobi equations on triangular meshes, *SIAM Journal on Scientific Computing*, 24(2003), 1005-1030.

- [89] J. Zhang, W. Klingsch, A. Schadschneider and A. Seyfried, Transitions in pedestrian fundamental diagrams of straight corridors and T-junctions, *J. Stat. Mech.*, P06004, 2011.
- [90] J. Zhang, W. Klingsch, A. Schadschneider and A. Seyfried, Ordering in bidirectional pedestrian flows and its influence on the fundamental diagram, *J. Stat. Mech.*, 02002, 2012.
- [91] J. Zhang, W. Klingsch, A. Schadschneider and A. Seyfried, Experimental study of pedestrian flow through a T-junction, *Traffic and Granular Flow '11*, 241–249, 2013.
- [92] H. Zhao, A Fast Sweeping method for Eikonal equations, *J. Math. and Computing*, vol 74, no 250, pp.603-627, 2005.





# CURRICULUM VITAE

## Personal Information

Name Raghavender Etikyala

## Education

<b>2001-2004</b>	Bachelor of Science
<b>2004-2006</b>	Master of Science in Mathematics
<b>2007-2009</b>	Master of Technology in Industrial Mathematics
<b>2010-</b>	Doctorate study in Mathematics at TU Kaiserslautern



# WISSENSCHAFTLICHER WERDEGANG

## Persönliche Informationen

Name Raghavender Etikyala

## Schulbildung

**2001-2004** Bachelor of Science

**2004-2006** Master of Science in Mathematik

**2007-2009** Master of Technology in Industriell Mathematik

**2010-** Doktorand der Mathematik an der TU Kaiserslautern